MCDONNELL DOUGLAS

PERFORMANCE ANALYSIS AND DESIGN SYNTHESIS (PADS) COMPUTER PROGRAM

VOLUME II

Program Description

Part 1

Final Report

NOVEMBER 1972 MDC G4287

PREPARED UNDER CONTRACT NO. NAS9-12059
BY GUIDANCE AND FLIGHT MECHANICS DEPARTMENT,
RESEARCH AND DEVELOPMENT
MCDONNELL DOUGLAS ASTRONAUTICS COMPANY—WEST
HUNTINGTON BEACH, CALIFORNIA
FOR
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

MCDONNELL DOUGLAS ASTRONAUTICS COMPANY-WEST
5301 Bolsa Avenue, Huntington Beach, CA 92647

764

N73-18831

MCDONNELL DOUGLAS

PERFORMANCE ANALYSIS AND DESIGN SYNTHESIS (PADS) COMPUTER PROGRAM
VOLUME II
Program Description
Part 1
Final Report

NOVEMBER 1972

MDC G4287

PREPARED UNDER CONTRACT NO. NAS9-12059
BY GUIDANCE AND FLIGHT MECHANICS DEPARTMENT,
RESEARCH AND DEVELOPMENT
MCDONNELL DOUGLAS ASTRONAUTICS COMPANY—WEST
HUNTINGTON BEACH, CALIFORNIA
FOR
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

(NASA-CR-128678) PERFORMANCE ANALYSIS AND DESIGN SYNTHESIS (PADS) COMPUTER PROGRAM. VOLUME 2: PROGRAM DESCRIPTION, PART 1 Final (McDonnell-Douglas Astronautics Co.) 760 p HC CSCL 22A

N73-18831

Unclas G3/30 50668

MCDONNELL DOUGLAS ASTRONAUTICS COMPANY-WEST

5301 Bolsa Avenue, Huntington Beach, CA 92647

760

FOREWORD

This is the second of three volumes describing the Performance Analyses and Design Synthesis (PADS) computer program. This volume is devoted to programming and numerical techniques. Volume I contains a complete program formulation and Volume III is a users manual.

The development of PADS was conducted by McDonnell Douglas Astronautics Company at Huntington Beach, California, under NASA contract NAS 9-12059, under the cognizance of Mr. Robert Abel, NASA, MSC, Houston, Texas. The key MDAC personnel who formulated and programmed PADS are Messrs. Murray H. Rosenberg, John W. Hensley, and Michael Beach. Valuable programming assistance was given by Larry Ong, Fred Gangloff, and Sheldon Herman.

ABSTRACT

The Performance Analysis and Design Synthesis (PADS) computer program has a two-fold purpose. It can size launch vehicles in conjunction with calculus-of-variations optimal trajectories and can also be used as a general-purpose branched trajectory optimization program. In the former use, it has the Space Shuttle Synthesis Program as well as a simplified stage weight module for optimally sizing manned recoverable launch vehicles. For trajectory optimization alone or with sizing, PADS has two trajectory modules. The first trajectory module uses the method of steepest descent; the second employs the method of quasilinearization, which requires a starting solution from the first trajectory module.

Section 1
INTRODUCTION

The PADS computer program is actually a merger of four separate programs. Each of the separate programs, hereafter called modules, has its own function to perform. Above all of these modules is a main program which calls the appropriate modules into action whenever the problem at hand requires. This main program also controls auxiliary activities concerned mainly with reading in data and communicating data from one module to another. The main modules and auxiliary functions fit naturally into an overlay structure which permits loading this very large program into a moderate core size on either the CDC 6000 or UNIVAC 1108 computer. This overlay structure is illustrated in Section 2 of this volume. The basic modules and auxiliary functions of PADS are listed and defined below.

Basic Modules

Overlay	Name	Abbreviation	Purpose
(3, 0)	Steepest Descent	SD or TOPM	Optimize trajectory and staging approximately and provide starting solution for the quasi-linearization module
(4, 0)	Quasi-Linearization	QL or GROPE	Solve Euler-Lagrange boundary value problem (calculus of variations)
(6, 0)	Phase I Sizing	SIZE	Synthesize generalized two-stage launch vehicle (simple sizing)

Overlay	Name	Abbreviation	Purpose
(7, 0)	Space Shuttle Synthesis Program	SSSP	Synthesize two-stage manned recoverable launch vehicle
		Auxiliary Functions	
(1, 0)	INPUT Editor	INEDIT	Read NAMELIST type input and store input data on file for access by various modules
(2, 0)	First Input Scan	GEINP	Scan and prepare basic input data for SD and/or QL module execution
(3, 1)	Second Input Scan	SDINP	Scan boundary conditions and set up steepest descent module data
(5, 0)	Sizing Trajectory Interface	SIZIN	Alter weights and other physical data according to sizing computation in SSSP or SIZE module in preparation for trajectory computation

This volume is organized according to the structure of the program. The next section describes how the main program and auxiliary functions interact with and permit communication between modules. Descriptions of the coding in each module are given in later sections.

The coding documentation includes subroutine glossaries, listings, flow charts, and descriptions of purpose and logic flow. Where possible, subroutine descriptions are tied back to the formulation document.

The documentation of each module consists of the description of the executive program, followed by a set of common glossaries pertinent only to that module. This is followed, in turn, by alphabetically ordered subroutine descriptions.

The final section of this volume contains a discussion of ground rules for modifying the computer program.

PADS Main Program 2

CONTENTS

Section 2
OVERALL PADS PROGRAM ORGANIZATION

The main program of PADS is called PADS1. Its flow chart is presented in a figure, with listing and glossary of terms placed directly after. The listings given in this section are flow-listed to indicate transfers within each routine. The listings have been made for the CDC 6500 version of PADS, with some notes where appropriate to indicate differences in the UNIVAC version. A key difference between CDC and UNIVAC listings is that the CDC listing contains a complete picture of all common blocks, whereas the UNIVAC listing only has INCLUDE statements for the common blocks.

Following the PADS1 description is a diagram of the PADS overlay structure, and COMMON block glossaries and subroutine descriptions that are shared by different modules.

Program PADS

Purpose

Program PADS is a dummy main program used to call the main executive routine, PADS1.

Description

The CDC version of PADS defines the required logical units in its argument list. The UNIVAC version has no such requirement.

```
      PROGRAM PADS(INPUT=1002,OUTPUT=1002,TAPE6=OUTPUT,TAPE5=INPUT,     PADS
     *TAPE1=1002,TAPE12=1002,TAPE4=1002,TAPE10=TAPE12)                  AAA
      CALL PADS1                                                        PADS
      END                                                               PADS
```

20 OCT 72 6.01-46

2-4

SUBRØUTINE
PADS1

SUBRØUTINE
PADS1

SUBRØUTINE PADS1

Subroutine PADS1

Entry Points SIZERR, SDERR, QLERR

Purpose

PADS1 is the main executive routine for the PADS program.

Description

As the main executive routine, PADS1 calls in the various modules of the program. These modules, shown in the PADS OVERLAY STRUCTURE, are actually a block of coding headed by an executive routine. The way that PADS1 calls in a module is different in the UNIVAC system than with the CDC software. The CDC software requires calls to the OVERLAY routine to bring in the program module, whereas in the UNIVAC version, simple calls to the appropriate executive routine are used. (See program flow list.) Additional differences in the UNIVAC version of PADS1 exclude the use of blocked binary files* and the sharing of file buffers.** These aspects are not available or necessary on the UNIVAC machine. In the CDC version the routines OPENMS, READMS, and WRITMS are FORTRAN library routines*** that set up, read, and write on random access storage. The UNIVAC version of PADS has special versions of these routines that in turn call UNIVAC-MSC peculiar routines RINIT, RREAD, and RWRITE (random access drum storage and retrieval routines).

The CDC 6000 system of disc storage is entirely logical unit number-oriented whereas random storage on the UNIVAC-MSC system is not. Following is a chart showing how non-core storage is set up on the UNIVAC system.

Entry Points

The entry points SIZERR, SDERR, and QLERR are control error returns that are called from various points in the different modules to return control to the proper place in the executive routine (PADS1). Depending on flag setting, corrective action is taken or the next case of input data is requested.

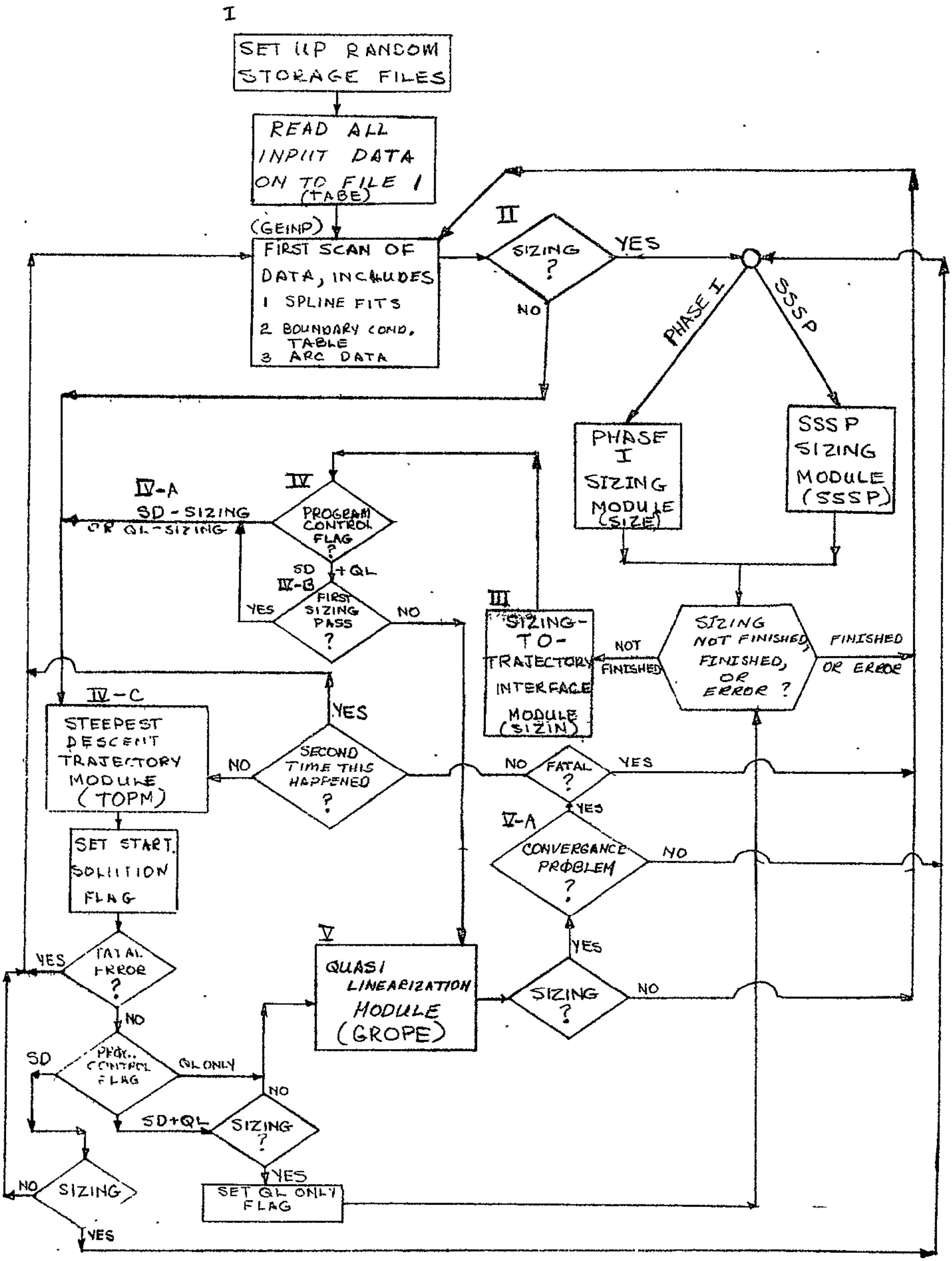
---

*CDC listing: CALL FTNBIN (1, 3, IBLKN). See CDC Scope 3.0 manual.
**CDC listing: CALL EQUBUFF (5LINPUT, 10, IIN, IFET, IEF)
***CDC FORTRAN Manual

UNIVAC Logical Unit Descriptions

Internal Logical Unit Number	Purpose	Type	Description
1	Input data file created by INEDIT	FASTRAND	Control card assigned as FASTRAND file 'A'
3	Scratch random file used by INEDIT	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
4	INEDIT scratch and Q-L Scratch	H.S. Drum Sequential	Automatically assigned to first half of drum
5	INPUT	Card Reader	
6	OUTPUT	Printer	
9	Arc data and boundary conditions	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
10	Scratch file used in TRAN3	FASTRAND	Assigned as FASTRAND file 'H'
11	Starting solution file	FASTRAND or Tape	Assigned as File I either tape (to save) or FASTRAND
12	Used in Q-L solution	H.S. Drum sequential	Automatically assigned to first half of drum
33	Bivariant aero coefficients set number 6	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
34	Bivariant aero, coefficients set number 7	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
35	Bivariant aero coefficients set number 8	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
36	Bivariant air breather engine data set number 9	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
39	Steepest descent trajectory storage	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
40	Steepest descent trajectory storage	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND
41	Steepest descent adjoint storage	H.S. Drum Random	Automatically assigned to second half of drum to overflow on to FASTRAND

PADS1


I
SET UP RANDOM STORAGE FILES
READ ALL INPUT DATA ON TO FILE 1 (TAPE)
(GEINP)
FIRST SCAN OF DATA, INCLUDES
1 SPLINE FITS
2 BOUNDARY COND. TABLE
3 ARC DATA
II
SIZING ?
YES
NO
PHASE I
SSSP
PHASE I SIZING MODULE (SIZE)
SSSP SIZING MODULE (SSSP)
IV-A
SD-SIZING OR QL-SIZING
IV
PROGRAM CONTROL FLAG ?
SD +QL
IV-B
FIRST SIZING PASS ?
YES
NO
III
SIZING-TO-TRAJECTORY INTERFACE MODULE (SIZIN)
NOT FINISHED
SIZING NOT FINISHED, FINISHED, OR ERROR ?
FINISHED OR ERROR
IV-C
STEEPEST DESCENT TRAJECTORY MODULE (TOPM)
YES
SECOND TIME THIS HAPPENED ?
NO
FATAL ?
YES
V-A
CONVERGANCE PROBLEM ?
SET START SOLUTION FLAG
FATAL ERROR ?
V
QUASI LINEARIZATION MODULE (GROPE)
SIZING ?
SD
PROG. CONTROL FLAG
QL ONLY
SD+QL
SIZING ?
SET QL ONLY FLAG
SIZING ?

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CAR		D	Constant, 1715.4827	/DATA /(	8)	PADS1	D	CAR
ER	E_R	I	Earth radius. (FT)	/GLOBAL/(	2)	COORDS	I	ER
						CRASH	I	REM
						EQUA3	I	ER
						GEINP	I	ER
						PADS1	I	ER
						PDBC	I	ER
						SOMG	I	ER
						TRTOSZ	I	ER
FTNM		D	Feet to naut. mi. conversion, 1.645791629×10^{-4}	/DATA /(	7)	OUT	I	FTNM
						PADS1	D	FTNM
						TRTOSZ	I	FTNM
GM	GM	O	Product of Newton's universal gravitational constant and the mass of the earth. (FT^3/SEC^2)	/GLOBAL/(	67)	CRASH	I	GM
						OUT	I	GM
						PADS1	O	GM
						PDBC	I	GM
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
ICONVR		I	A seven word array not used.	/GLOBAL/(	96)	PADS1	I	ICONVR
ID		D	A four word array containing the basic deck, reference run, case and part case numbers in that order.	/GLOBAL/(	21)	BLICO	I	ID
						FRENCH	I	ID
						GEINP	I	ID
						PADS1	D	ID
						PRINT	I	ID
						SDINP	I	ID
						TOPM	I	ID
						VEHDF	I	ID
IFATAL		I	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADS1	I	IFATAL
						SDINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
IFLOP		W	Q L convergance failure counter	/PADS1 /(*	)	PADS1	W	IFLOP
KSOL		O	An internal flag that has the same significance as ITPSO.	/GLOBAL/(	94)	FNTG	I	KSOL
						PADS1	O	KSOL
						SDINP	M	KSOL
LUM		M	Program control flag. LUM = 0: Steepest descent only; LUM = 1: Steepest descent and adjoint transformation stored on tape; LUM = 2: Steepest descent and QL; LUM = 3: QL only.	/GLOBAL/(	6)	AST3	I	LUM
						FNTG	I	LUM
						GEINP	I	LUM
						PADS1	M	LUM
						SDINP	I	LUM
						TOPM	M	LUM
PI	π	D	Constant 3.141592653	/DATA /(	1)	OUT	I	PI
						PADS1	D	PI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
RAD		D	Radian to angle conversion, 57.29577951	/DATA /(	2)	BEROCO	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADS1	D	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD
RDI		D	Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	D	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI
SC		D	Constant in Sutherlands equation, 198	/DATA /(	4)	PADS1	D	SC
						PAT63	I	SC
TMPF		D	Constant in Sutherlands equation, 392	/DATA /(	6)	PADS1	D	TMPF
						PAT63	I	TF
TZDD		D	Booster liftoff weight (lb)	/SIZING/(	1)	PADS1	D	TZDD
						SIZIN	I	TZ
						TAMPER	0	TZ
UMF		D	Constant in Sutherlands equation, $.301 \times 10^{-6}$	/DATA /(	5)	PADS1	D	UMF
						PAT63	I	UMUF

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CAR		O	Constant, 1715.4827	/DATA /(	8)	PADS1	O	CAR
ER	E_R	I	Earth radius. (FT)	/GLOBAL/(	2)	COORDS	I	ER
						CRASH	I	REM
						EQUA3	I	ER
						GEINP	I	ER
						PADS1	I	ER
						PDBC	I	ER
						SOMG	I	ER
						TRTOSZ	I	ER
FTNM		O	Feet to naut. mi. conversion, $1.645791629 \times 10^{-4}$	/DATA /(	7)	OUT	I	FTNM
						PADS1	O	FTNM
						TRTOSZ	I	FTNM
GM	GM	O	Product of Newton's universal gravitational constant and the mass of the earth. (FT^3/SEC^2)	/GLOBAL/(	67)	CRASH	I	GM
						OUT	I	GM
						PADS1	O	GM
						PDBC	I	GM
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
ICONVR		I	A seven word array not used.	/GLOBAL/(	96)	PADS1	I	ICONVR
ID		O	A four word array containing the basic deck, reference run, case and part case numbers in that order.	/GLOBAL/(	21)	BLICO	I	ID
						FRENCH	I	ID
						GEINP	I	ID
						PADS1	O	ID
						PRINT	I	ID
						SDINP	I	ID
						TOPM	I	ID
						VEHDF	I	ID
IFATAL		I	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADS1	I	IFATAL
						SDINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
IFLOP		W	Q L convergance failure counter	/PADS1 /(*	)	PADS1	W	IFLOP
IPASS		M	Sizing iteration counter	/SIZING/(	291)	GEINP	O	IPASS
						PADS1	M	IPASS
						PAYO2	I	IPASS
						SIZE	M	IPASS
						SIZIN	I	IPASS
						SSSP	M	IPASS
JTYP		I	Sizing. Flag.	/SIZING/(	313)	FNTG	I	JTYP
						GEINP	O	JTYP
						MODELA	I	JTYP
						PADS1	I	JTYP
						PROPIN	I	JTYP
						SIZIN	I	JTYP
						TRTOSZ	I	JTYP
KSOL		O	An internal flag that has the same significance as ITPSO.	/GLOBAL/(	94)	FNTG	I	KSOL
						PADS1	O	KSOL
						SDINP	M	KSOL

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
LUM		M	Program control flag. LUM = 0: Steepest descent only; LUM = 1: Steepest descent and adjoint transformation stored on tape; LUM = 2: Steepest descent and QL; LUM = 3: QL only.	/GLOBAL/(	6)	AST3 FNTG GEINP PADS1 SDINP TOPM	I I I M I M	LUM LUM LUM LUM LUM LUM
PI	π	D	Constant 3.141592653	/DATA /(	1)	OUT PADS1	I D	PI PI
RAD		D	Radian to angle conversion, 57.29577951	/DATA /(	2)	BEROCD BLGCON ENVFRM EQUA3 FNTG GUI3A MODELA MTX3A OUT PADS1 SDINP TRTOSZ	I I I I I I I I I D I I	DEG RAD RAD RAD RAD RAD RAD RAD RAD RAD RAD RAD
RDI		D	Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO DER3A FNTG GUI3A MODELA MODELB PADS1 PROPB PROPIN REU3 SDINP SOMG	I I I I I I D I I I I I	RDI RDI RDI RDI RDI RDI RDI RDI RDI RDI RDI RDI
SC		D	Constant in Sutherlands equation, 198	/DATA /(	4)	PADS1 PAT63	D I	SC SC
TMPF		D	Constant in Sutherlands equation, 392	/DATA /(	6)	PADS1 PAT63	D I	TMPF TF
TRAFLG		I	Traffic control flag 0. Sizing loop not converged 1. Sizing loop converged 2. Error in sizing loop next case	/SIZING/(	284)	FRENCH ITER8 PADS1 SIZE SSSP VEHOF	O O I M O O	TRAFLG TRAFLG TRAFLG TRAFLG TRAFLG TRAFLG
UMF		D	Constant in Sutherlands equation, $.301 \times 10^{-6}$	/DATA /(	5)	PADS1 PAT63	D I	UMF UMUF

```
      SUBROUTINE PADS1                                                  PADS1
      DIMENSION IIN(10),IFET(170),I39(20),I40(20),I41(60),I9(24),       PADS1
     *IBLKN(3),I33(182),I34(182),I35(182)                               AAA
     *,I36(182)                                                         POW
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      , ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      , ARCDAT
     *XLMAX     ,HDMAX     ,BMDOT     ,ALFMAX    ,PHMAX     ,MAEA      , ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      , ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      , ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        , ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                  ARCDAT
     * ,FRATE   , ARCD(9)                                               RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      COMMON/BICUBE/  BIC(273)                                          PADS1
      COMMON/DATA/                                                      DATA
     *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      , DATA
     *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4      , DATA
      DATA          PI,RAD,RDI,SC,UMF,TMPF,FTNM,CAR/                    UH
     1                   3.141592653,57.29577951,.01745329252,198.,     UH
     * .302E-6,392.0 , 1.645791629E-4 , 1715.4827 /                     UH
      COMMON/ ORBIT/  VI,       GAMI,      PSII,      XMUI,        P,   ORBIT
     *      ECC,     AINCL,      ARGP,    ASCNOD,    SMIMAJ,   APOGEE,  ORBIT
     *  PERGEE,   ANOMLY,       CAPX,      CAPY,     ASYMP,   ENERGY,   ORBIT
     *   HMNTM,    DVIDV,     DVIDG,                                    ORBIT
     *  DVIDH ,    DVIDM,    DVIDPS,    DVIDRO,    DVIDMU,    DGIDV,    ORBIT
     *  DGIDG ,    DGIDH,    DGIDM ,    DGIDPS,    DGIDRO,    DGIDMU,   ORBIT
     *  DPIDV ,    DPIDG,    DPIDH ,    DPIDM ,    DPIDPS,    DPIDRO,   ORBIT
     *  DPIDMU,    DMIDV,    DMIDG ,    DMIDH ,    DMIDM ,    DMIDPS,   ORBIT
     *  DMIDRO,    DMIDMU,   DPDV  ,    DPDG  ,    DPDH  ,    DPDM  ,   ORBIT
     *  DPDPS ,    DPDRO ,   DPDMU ,    DECDV ,    DECDG ,    DECDH ,   ORBIT
     *  DECDM ,    DECDPS,   DECDRO,    DECDMU,    DIDV  ,    DIDG  ,   ORBIT
     *  DIDH  ,    DIDM  ,  DIDPS  ,   DIDRO  ,   DIDMU  ,   DBEDV  ,   ORBIT
     *  DBEDG ,    DBEDH ,   DBEDM ,    DBEDPS,    DBEDRO,    DBEDMU,   ORBIT
     *  DNODV ,    DNODG ,   DNODH ,    DNODM ,    DNODPS,    DNODRO,   ORBIT
     *  DNODMU,    DSMDV,     DSMDG,     DSMDH,     DSMDM,    DSMDPS    ORBIT
      COMMON/ORBIT/                                                     ORBIT
     *  DSMDRO,   DSMDMU,    DAPDV,     DAPDG,     DAPDH,    DAPDM ,    ORBIT
     *  DAPDPS,   DAPDRO,    DAPDMU,    DPEDV,     DPEDG,    DPEDH ,    ORBIT
     *DPEDM     ,DPEDPS    ,DPEDRO    ,DPEDMU    ,DANDV     ,DANDG     , ORBIT
     *  DANDH ,   DANDM ,    DANDPS,    DANDRO,    DANDMU,   DCXDV ,    ORBIT
     *  DCXDG ,   DCXDH ,    DCXDM ,    DCXDPS,    DCXDRO,   DCXDMU,    ORBIT
     *  DCYDV ,   DCYDG ,    DCYDH ,    DCYDM ,    DCYDPS,   DCYDRO,    ORBIT
     *  DCYDMU,   DASDV ,    DASDG ,    DASDH ,    DASDM ,   DASDPS,    ORBIT
     *  DASDRO,   DASDMU,    DENDV ,    DENDG ,    DENDH ,   DENDM ,    ORBIT
     *  DENDPS,   DENDRO,    DENDMU,    DMODV ,    DMODG ,   DMODH ,    ORBIT
     *  DMODM ,   DMODPS,    DMODRO,    DMODMU                          ORBIT
      DIMENSION ORBPRM(18),PPO(7,18)                                    ORBIT
      EQUIVALENCE (VI,ORBPRM),(DVIDV,PPO)                               ORBIT
      COMMON/ORBIT/ YMXRF,SNXLMR,CSXLMR,SDOWN,SCROSS,TD,TC              ORBIT
     *,SNPSR    ,CSPSR     ,SNGI      ,CSGI      ,SPSII     ,CPSII     , ORBIT
     * STOT     ,CSI       ,SNI       ,SNGNU     ,CSANO     ,COSDMU    , ORBIT
     * SINDMU   ,THT       ,WTFUEL                                      ORBIT
      COMMON/GLOBAL/                                                    GLOBAL
     *GR        ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM        GLOBAL
     *,JJOP(10) ,IFATAL    ,NARC      ,NBRAN     ,NFARC     ,ID(4)      GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                              GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEDFL(20)     GLOBAL
     *,ITPSD ,KSOL     ,KGLOBL(8)                                       RETAP
      LOGICAL IFATAL                                                    PADS1
      EQUIVALENCE(ICONVR,KGLOBL(2))                                     AAA
      REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                           SIZING
      COMMON /SIZING/                                                   SIZING
C           PHASE II SIZING PARAMERERS                                  SIZING
     *TZ,       VV(3),     QP(14),    EROR,     PZ(5),     VQ,     SW(20), SIZING
     *SV(28),   SQ(37,5),  SE(11),    TLAT,     TLNG,                   SIZING
C          PHASE I SIZING PARAMERERS                                    SIZING
     *WBO,      WLDO,      DWEB,      DWEO,      TOLWT,    WPB,    TWRAT2, SIZING
     *BK1,      BK2,       BK3,       BK4,       ISIZE,    TRAFLG, TWRATO, SIZING
     *DK1,      DK2,       DK3,       DK4,       PRFLG,    IPASS,  IPSMAX, SIZING
     *AEXIT,    TVACO,     NO,        WFO,       IDVEL,    ISPO,   ISPB,   SIZING
     *XPL,      TVACB,     NNB,       WEO,       WEB,      WO,     WLO,    SIZING
     *DVO,      DVB,       MUB,       MUO,       VSTG,     WPD             SIZING
     *,JTYP,    BECO ,     BSTG ,     ORBI,      ITNBW , ITNOW  ,          SIZING
```

```
     * SVOPSQ   , SVOCON  ,IHUNT    ,IOPSTG  ,ISZO(19)                  UH
       DIMENSION TZOO(101)                                               PADS1
       COMMON/POWEL/ ITZLE(156)                                          OPTSTG
       EQUIVALENCE(TZOO,TZ)                                              PADS1
       DATA TZOO/101*0/                                                  PADS1
       DIMENSION PT(5)                                                   AAA
C              MAIN  PROGRAM FOR  PADS  -NASA HOUSTON                    PADS1
C      WRITTEN BY  MCDONNELL-DOUGLAS  ASTRONAUTICS  COMPANY              PADS1
C                  HUNTINGTON  BEACH  CALIFORNIA    1972                 PADS1
C                                                                        PADS1
C       I   SET-UP  STORAGE FILES                                        PADS1
       COMMON/TABLE/ TABLE(2100)                                         PADS1
       COMMON/LASTAB/ LASTAB,IPCT(33),IPPNTR,NTAPE,MAXPCT,BLANK          AAA
       DATA  ID / 4*0 /                                                  PADS1
       DATA  IPPNTR,NTAPE,BLANK,MAXPCT / 0,1,4H     ,33 /                AAA
       IIN(1) =6LTAPE39                                                  PADS1
       IIN(2) =6LTAPE40                                                  PADS1
       IIN(3) =6LTAPE41                                                  PADS1
       IIN(4) =6LTAPE11                                                  PADS1
       IIN(5)=6LTAPE36                                                   POW
       IIN(6) =5LTAPE9                                                   PADS1
       IIN(7)=6LTAPE33                                                   PADS1
       IIN(8)=6LTAPE34                                                   PADS1
       IIN(9)=6LTAPE35                                                   PADS1
       IIN(10)=5LTAPE3                                                   AAA
       CALL EQUBUFF(5LINPUT,10,IIN,IFET,IEF)                             PADS1
       IF(IEF.EQ.0) GO TO 10                                             PADS1
       CALL  STPIT(88)                                                   PADS1
   10 CONTINUE                                                           PADS1
       CALL OPENMS(33,I33,182,0)                                         PADS1
       CALL OPENMS(34,I34,182,0)                                         PADS1
       CALL OPENMS(35,I35,182,0)                                         PADS1
       CALL OPENMS(36,I36,182,0)                                         POW
       CALL OPENMS(39,I39,20,0)                                          PADS1
       CALL OPENMS(40,I40,20,0)                                          PADS1
       CALL OPENMS(41,I41,60,0)                                          PADS1
       CALL OPENMS(9, I9, 22,0)                                          PADS1
       CALL OPENMS(3,PT,5,0)                                             AAA
       IBLKN(1)=1                                                        PADS1
       IBLKN(2)=12                                                       AAA
       IBLKN(3)=4                                                        PADS1
       CALL FTNBIN(1,3,IBLKN)                                            PADS1
       CALL OVERLAY(4HTABI,1,0,0)                                        PADS1
   20 CALL OVERLAY(4HGEIN,2,0,0)                                         PADS1
       GR=GR*ER*ER                                                       PADS1
C       II   TEST FOR SIZING                                             PADS1
       IFLOP =0                                                          JULY20
       IF(JTYP .EQ.0) GO TO  100                                         PADS1
    5 GO TO (11,21),JTYP                                                 PADS1
   11 CALL OVERLAY(4HPHS1,6,0,0)                                         PADS1
       GO TO 50                                                          PADS1
   21 CALL OVERLAY(4HSSSP,7,0,0)                                         PADS1
       ENTRY SIZERR                                                      AAA
   50 IF(TRAFLG-1.) 51,300,20                                            UH
C       III     SET-UP  DATA FOR TRAJECTORY                              PADS1
   51 CALL OVERLAY(5HSIZIN ,5,0,0)                                       PADS1
C       IV     QL OR SD  TRAJECTORY                                      PADS1
       IF(LUM-2) 60,80,100                                               PADS1
C       IV-A   STEEPEST  DESCENT  SIZING                                 PADS1
   60 LUM =1                                                             PADS1
       GO TO  100                                                        PADS1
C       IV-B   QL ALL THE WAY  TEST FOR  FIRST SIZING PASS               PADS1
   80 IF(IPASS-2)  100,100,200                                           PADS1
C       IV-C   STEEPEST DESCENT TRAJECTORY                               PADS1
  100 CALL OVERLAY(5HITOPM,3,0,0)                                        PADS1
       ENTRY SDERR                                                       PADS1
       KSOL=1                                                            PADS1
       IF(IFATAL) GO TO  20                                              PADS1
       IF(LUM -2) 110, 115,200                                           PADS1
  110 IF(JTYP .GT.0) GO TO 5                                             PADS1
```

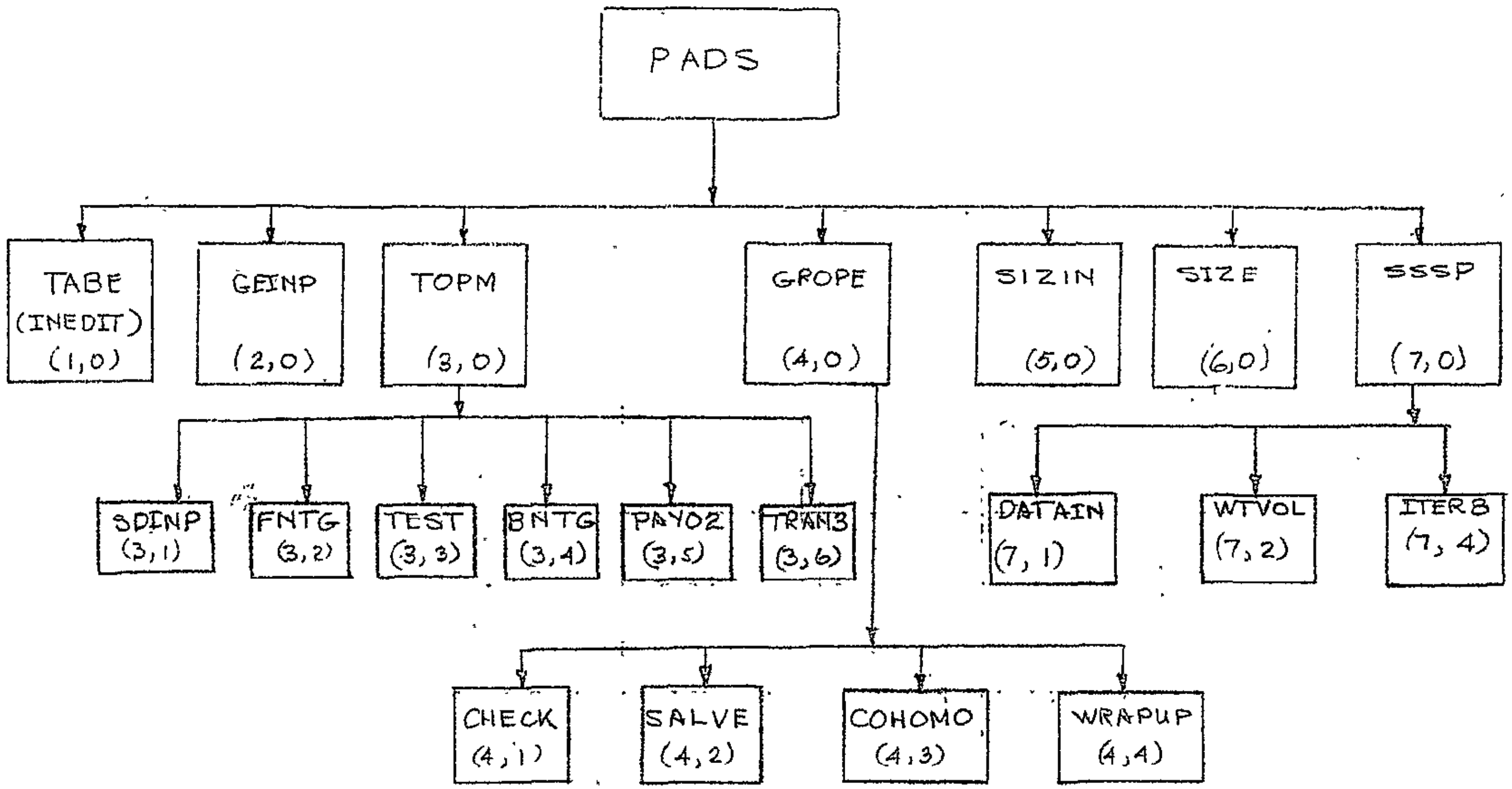
10 100 11 21 50 51 300 20 60 80 100 100 200 100 20 110 115 200 5

20 OCT 72 G.01-46

```
      GO TO  20                                                      PADS1
  115 IF(JTYP .EQ.0) GO TO 200                                       PADS1
      LUM=3                                                          PADS1
      GO TO 50                                                       PADS1
C       V    QUASILINEARIZATION  SOLUTION                            PADS1
  200 CALL OVERLAY(5HGROPE,4,0,0)                                    PADS1
C       V-A    TEST FOR CONVERGANCE FAILURE  AND SIZING FLAG         PADS1
      ENTRY QLERR                                                    PADS1
      IF(JTYP .EQ.0) GO TO 20                                        PADS1
      IF(ICONVR.EQ.0) GO TO 5                                        PADS1
C       V-B    QL-CONVERGANCE PROBLEM  RETURN TO SD                  PADS1
      LUM =2                                                         PADS1
      IF(IFATAL) GO TO  20                                           PADS1
      IFLOP =IFLOP+1                                                 JULY28
      IF(IFLOP.GT.1) GO TO 20                                        JULY28
      IF(IPASS.LE.2) IPASS=3                                         JULY28
      GO TO 100                                                      PADS1
C     VI     TEST FOR  HUNTING                                       PADS1
  300 GO TO 20                                                       UH
      END                                                            PADS1
```

20 OCT 72 G.01-46

PADS OVERLAY STRUCTURE



BLØCK
ARCDAT

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBR	CODE	VAR
SREF	S_{ref}	Aerodynamic reference area	(FT^2)	/ARCDAT/(	1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SOINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF
EJ	A_{exit}	Nozzle exit area	(FT^2)	/ARCDAT/(	2)	PROPB	I	EJ
						PROPIN	I	EJ
						SIZIN	O	EJ
XISP	I_{sp}	Vacuum specific impulse	(SEC)	/ARCDAT/(	3)	IMPUL	I	XISP
						SIZIN	O	XISP
TMULT	T_{mult}	Thrust multiplier or number of engines		/ARCDAT/(	4)	EQUA3	I	TMULT
						FXDAT	M	TMULT
						PROPB	I	TMULT
						PROPIN	I	TMULT
						SIZIN	O	TMULT
DTNC	$\Delta\tau$	Integration interval	(SEC)	/ARCDAT/(	5)	BNTG	I	DTNC
						FNTG	I	DTNC
						GEINP	M	DTNC
						PROPIN	I	DTNC
DTPI		Print frequency for trajectory		/ARCDAT/(	6)	FNTG	I	DTPI
						GEINP	M	DTPI
IATM		Atmosphere option flag		/ARCDAT/(	7)	EQUA3	I	IATM
						FXDAT	I	IATM
						OUT	I	IATM
						PDBC	I	IATM
						VT	I	IATM
IMOD		Control mode option flag		/ARCDAT/(	8)	FXDAT	M	IMODE
						SOINP	I	IMODE
JAER		Aerodynamic model option flag		/ARCDAT/(	9)	BEROCO	I	JAER
						EQUA3	I	JAER
						GEINP	I	JAER
						OUT	I	JAER
						PROPB	I	JAER
						PROPIN	I	JAER
						VT	I	JAER
JPRO		Propulsion model option flag		/ARCDAT/(	10)	EQUA3	I	JPRO
						GEINP	I	JPRO
						IMPUL	I	JPRO
						MODELA	I	JPRO
						PROPB	I	JPRO
						PROPIN	I	JPRO
QMAX	q_{MAX}	Maximum dynamic pressure instantaneous inequality limit	(PSF)	/ARCDAT/(	11)	PROPB	I	QMAX
GMAX	G_{MAX}	Maximum total acceleration g load		/ARCDAT/(	12)	BL5	I	GMAX
						FH3	I	GMAX
						MODELA	I	GMAX
						PROPB	I	GMAX
						PROPIN	I	GMAX
XLAMX	L_{MAX}	Maximum aerodynamic lift	(LBS)	/ARCDAT/(	13)	MODELA	I	XLMAX
HDMAX	$\dot{Q}_{MAX}$	Maximum heating rate inequality constraint		/ARCDAT/(	14)	PROPB	I	HDMAX
GMDOT	$\dot{\gamma}*$	Pitch rate	(DEG/SEC)	/ARCDAT/(	15)	DER3A	I	GMDOT
						MODELA	I	GMDOT
						MODELB	I	GMDOT
						PROPB	I	GMDOT
						PROPIN	I	GMDOT

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALFMAX	α_{MAX}	Maximum angle of attack (DEG)	/ARCDAT/(	16)	INBVAD	M	ALFMAX
					MODELA	I	ALFMAX
					MODELB	I	ALFMAX
PHMAX		Belly down flag	/ARCDAT/(	17)			
MAEA		Curve number	/ARCDAT/(	18)	EQUA3	I	MAEA
					GEINP	I	MAEA
					PROPB	I	MAEA
					PROPIN	I	MAEA
MAEB		Curve number	/ARCDAT/(	19)	EQUA3	I	MAEB
MAEC		Curve number	/ARCDAT/(	20)	EQUA3	I	MAEC
MAED		Curve number	/ARCDAT/(	21)	EQUA3	I	MAED
MAEE		Curve number	/ARCDAT/(	22)	EQUA3	I	MAEE
MAEF		Curve number	/ARCDAT/(	23)	EQUA3	I	MAEF
MAEG		Curve number	/ARCDAT/(	24)	EQUA3	I	MAEG
MT		Curve number -thrust table	/ARCDAT/(	25)	FXDAT	I	MT
					PROPB	I	MT
					PROPIN	I	MT
					THRUST	I	MT
MISP		Curve number KISP loss table	/ARCDAT/(	26)	IMPUL	I	MISP
					PROPB	I	MISP
					PROPIN	I	MISP
MAXCG		Curve number -xcg table	/ARCDAT/(	27)	EQUA3	I	MXCG
MZCG		Curve number- zcg table	/ARCDAT/(	28)	EQUA3	I	MZCG
MWDB		Description not input	/ARCDAT/(	30)	BLICO	I	MWDB
					GEINP	I	MWDB
					PROPB	I	MWDB
					PROPIN	I	MWDB
MDB		Curve number - base drag table	/ARCDAT/(	31)	EQUA3	I	MDB
XCGR	X_{CGR}	Reference xcg location (FT)	/ARCDAT/(	32)	VT	I	XCGR
ZCGR	Z_{CGR}	Reference zcg location (FT)	/ARCDAT/(	33)	VT	I	ZCGR
XE	X_E	Engine thrust centroid body x station	/ARCDAT/(	34)	EL2	I	XE
ZE	Z_E	Engine thrust centroid body z station	/ARCDAT/(	35)	EL2	I	ZE
					VT	I	ZE
XT	X_T	Aerodynamic trim surface body x station	/ARCDAT/(	36)	VT	I	XT
DREF	D_{ref}	Aerodynamic reference length	/ARCDAT/(	37)	VT	I	DREF
RHOB	ρ_b	Atmosphere base density for heating calculation (LB/FT**3)	/ARCDAT/(	39)	DER3A	I	RHOB
					FXDAT	O	RHOB
					PDY3A	I	RHOB
QMULT	=0 OR 1	Heating flag multiplier	/ARCDAT/(	40)	DER3A	I	QMULT
					FXDAT	O	QMULT
					PDY3A	I	QMULT
REMAX	$R_{ey_{MAX}}$	Maximum unit reynolds number inequality constraint	/ARCDAT/(	41)	PROPB	I	REMAX
FRATE		Input rated vacuum thrust per engine (LBS)	/ARCDAT/(	42)	EQUA3	I	FRATE
					FXDAT	I	FRATE
					PROPB	I	FRATE
					PROPIN	I	FRATE
					SIZIN	O	FRATE

BLØCK
BICUBE

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AMIN	x_0	The smallest value of the first independent variable of a bivariate table.	/BICUBE/(	1)	BLICO BLYNE INBVAD	M I I	MMIN AMIN AMIN
AMAX	x_N	The largest value of the first independent variable of a bivariate table.	/BICUBE/(	2)	BLICO BLYNE INBVAD	O I I	MMAX AMAX AMAX
IF		Last file in the grid in which interpolation occurred.	/BICUBE/(	3)	BLYNE INBVAD	M M	IF IF
IFMAX	N	Total number of files in grid.	/BICUBE/(	4)	BLICO BLYNE INBVAD	M I I	IRMAX IFMAX IFMAX
MMIN	y_0	The smallest value of the second independent variable of a bivariate table.	/BICUBE/(	5)	BLICO BLYNE	O I	AMIN MMIN
MMAX	y_M	The largest value of the second independent variable of a bivariate table.	/BICUBE/(	6)	BLICO BLYNE	O I	AMAX MMAX
IR		Last rank in the grid in which interpolation occurred.	/BICUBE/(	7)	BLICO BLYNE INBVAD	M M M	IF IR IR
IRMAX		Total number of ranks in grid.	/BICUBE/(	8)	BLICO BLYNE INBVAD	M I I	IFMAX IRMAX IRMAX
IUNIT		Logical unit number on which bicubic spline coefficients are stored for this table.	/BICUBE/(	9)	BLICO BLYNE INBVAD	M I I	IUNIT IUNIT IUNIT
IRECT		Grid rectangle associated with IR and IF.	/BICUBE/(	10)	BLICO BLYNE INBVAD	M M M	IRECT IRECT IRECT
IREC		Logical record on IUNIT that contains spline coefficients for rectangle IRECT.	/BICUBE/(	11)	BLICO BLYNE INBVAD	M M M	IREC IREC IREC
C		A 32 word array containing the spline coefficients for the two bivariate functions at rectangle IRECT.	/BICUBE/(	12)	BLYNE BLYNE INBVAD	O I O	C CL00 C
CL10		Description not input	/BICUBE/(	13)	BLYNE	I	CL10
CL20		Description not input	/BICUBE/(	14)	BLYNE	I	CL20
CL30		Description not input	/BICUBE/(	15)	BLYNE	I	CL30
CL01		Description not input	/BICUBE/(	16)	BLYNE	I	CL01
CL11		Description not input	/BICUBE/(	17)	BLYNE	I	CL11
CL21		Description not input	/BICUBE/(	18)	BLYNE	I	CL21
CL31		Description not input	/BICUBE/(	19)	BLYNE	I	CL31
CL02		Description not input	/BICUBE/(	20)	BLYNE	I	CL02
CL12		Description not input	/BICUBE/(	21)	BLYNE	I	CL12
CL22		Description not input	/BICUBE/(	22)	BLYNE	I	CL22
CL32		Description not input	/BICUBE/(	23)	BLYNE	I	CL32
CL03		Description not input	/BICUBE/(	24)	BLYNE	I	CL03
CL13		Description not input	/BICUBE/(	25)	BLYNE	I	CL13
CL23		Description not input	/BICUBE/(	26)	BLYNE	I	CL23
CL33		Description not input	/BICUBE/(	27)	BLYNE	I	CL33
CD00		Description not input	/BICUBE/(	28)	BLYNE	I	CD00
CD10		Description not input	/BICUBE/(	29)	BLYNE	I	CD10
CD20		Description not input	/BICUBE/(	30)	BLYNE	I	CD20
CD30		Description not input	/BICUBE/(	31)	BLYNE	I	CD30
CD01		Description not input	/BICUBE/(	32)	BLYNE	I	CD01
CD11		Description not input	/BICUBE/(	33)	BLYNE	I	CD11
CD21		Description not input	/BICUBE/(	34)	BLYNE	I	CD21
CD31		Description not input	/BICUBE/(	35)	BLYNE	I	CD31

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CD02		Description not input	/BICUBE/(	36)	BLYNE	I	CD02
CD12		Description not input	/BICUBE/(	37)	BLYNE	I	CD12
CD22		Description not input	/BICUBE/(	38)	BLYNE	I	CD22
CD32		Description not input	/BICUBE/(	39)	BLYNE	I	CD32
CD03		Description not input	/BICUBE/(	40)	BLYNE	I	CD03
CD13		Description not input	/BICUBE/(	41)	BLYNE	I	CD13
CD23		Description not input	/BICUBE/(	42)	BLYNE	I	CD23
CD33		Description not input	/BICUBE/(	43)	BLYNE	I	CD33
T		A 160 word array containing logical record IREC.	/BICUBE/(	44)	BLICO	I	T
					BLYNE	I	T
					INBVAD	I	T
ALFA		A 31 word array containing the mesh $x_0, x_1, \ldots, x_n$	/BICUBE/(	204)	BLICO	I	F
					BLICO	M	MACH
					BLYNE	I	ALFA
MACH		A 31 word array containing the mesh $y_0, y_1, \ldots, y_n$	/BICUBE/(	235)	BLICO	M	ALFA
					BLYNE	I	MACH

BLØCK
GLØBAL

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
GR	g_r	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
					BL5	I	GR
					EQUA3	I	GR
					FH3	I	GR
					GEINP	I	G
					GEINP	I	GR
					GEINP	O	IG
					OUT	I	GR
					PADS1	I	GR
					PDBC	I	GR
					REU3	I	GR
					SDINP	I	GR
					SIZE	I	GR
					SIZ1	I	GR
					SIZ2	I	GR
					SIZ3	I	GR
					SIZ4	I	GR
					SOMG	I	GR
					STAU	I	GR
ER	E_R	Earth radius. (FT)	/GLOBAL/(	2)	COORDS	I	ER
					CRASH	I	REM
					EQUA3	I	ER
					GEINP	I	ER
					PADS1	I	ER
					PDBC	I	ER
					SOMG	I	ER
					TRTDSZ	I	ER
OMGZ	ω	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
					CRASH	I	OMEGA
					DER3A	I	OMGZ
					EQUA3	I	OMGZ
					GEINP	I	OMGZ
					MODELA	I	OMGZ
					MODELB	I	OMGZ
					PDBC	I	OMGZ
					PDY3A	I	OMGZ
					SDINP	I	OMGZ
					TOPM	I	OMGZ
XLAMRF	ρ_r	Reference latitude. (DEG)	/GLOBAL/(	4)	CRASH	M	RHOO
					GEINP	I	XLAMRF
					REU3	I	XLAMRF
					SDINP	M	XLAMRF
YMURF	μ_r	Reference longitude. (DEG)	/GLOBAL/(	5)	CRASH	M	UMUO
					GEINP	I	YMURF
					REU3	I	YMURF
					SDINP	M	YMURF
LUM		Program control flag. LUM = 0: Steepest descent only; LUM = 1: Steepest descent and adjoint transformation stored on tape; LUM = 2: Steepest descent and QL; LUM = 3: QL only.	/GLOBAL/(	6)	AST3	I	LUM
					FNTG	I	LUM
					GEINP	I	LUM
					PADS1	M	LUM
					SDINP	I	LUM
					TOPM	M	LUM
TO	t_0	Trajectory start time. (SEC)	/GLOBAL/(	7)	SDINP	O	TINIT
EPSLON	ϵ	QL iteration convergence criterion.	/GLOBAL/(	8)			
INNER		Number of Adams-Moulton inner loops.	/GLOBAL/(	9)			
ITRMAX		Maximum number of QL iterations.	/GLOBAL/(	10)			
JJOP		A six word array used for various internal flags.	/GLOBAL/(	11)			
DONE		Description not input	/GLOBAL/(	12)	BLICO	M	DONE
IFATAL		Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
					GEINP	O	IFATAL
					PADS1	I	IFATAL
					SDINP	M	IFATAL
					SPLICD	M	IFATAL
					STPIT	O	IFATAL
					TOPM	M	IFATAL

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NARC	N_3	Number of subarcs in the problem.	/GLOBAL/(	18)	FNTG	I	NARC
					GEINP	M	NARC
					PROPIN	I	NARC
					SDINP	I	NARC
					SIZIN	I	NARC
NBRAN	N_1	Number of the last subarc on the stem of a branch problem. If the problem is not a branch problem, then NBRAN = 0.	/GLOBAL/(	19)	SDINP	O	NBRAN
NFARC	N_2	Number of the last subarc on the first branch. If the problem is not a branch problem, then NFARC = NARC.	/GLOBAL/(	20)	SDINP	O	NFARC
ID		A four word array containing the basic deck, reference run, case and part case numbers in that order.	/GLOBAL/(	21)	BLICO	I	ID
					FRENCH	I	ID
					GEINP	I	ID
					PADS1	O	ID
					PRINT	I	ID
					SDINP	I	ID
					TOPM	I	ID
					VEHDF	I	ID
KTAB		A 20 word array containing the number of state target conditions specified at the end of each subarc.	/GLOBAL/(	25)	GEINP	I	KTAB
					SDINP	I	KTAB
					SIZIN	I	KTAB
ITAB		A 20 word array containing the number of nonzero state initial conditions specified at the beginning of each subarc.	/GLOBAL/(	45)	GEINP	I	ITAB
					SDINP	I	ITAB
					SIZIN	I	ITAB
SIG		Payoff sign. SIG < 0: Payoff to be minimized; SIG > 0: Payoff to be maximized.	/GLOBAL/(	65)	PAY02	I	SIG
					SDINP	M	SIG
					TEST	I	SIG
					TRAN3	I	SIG
MAXTAB		Largest univariant table number in this case.	/GLOBAL/(	66)	SPLICO	M	NT
					SPLIZ	I	NT
					SPLYNE	I	NT
GM	GM	Product of Newton's universal gravitational constant and the mass of the earth. (FT^3/SEC^2)	/GLOBAL/(	67)	CRASH	I	GM
					OUT	I	GM
					PADS1	O	GM
					PDBC	I	GM
PSIRF	ψ_r	Reference azimuth. (DEG)	/GLOBAL/(	68)	GEINP	I	PSIRF
					REU3	I	PSIRF
					SDINP	M	PSIRF
IPFLG1		IPFLG1≠0 supresses print-out of velocity losses and inertial Euler angles.	/GLOBAL/(	69)	FNTG	I	IPFLG1
					OUT	I	IPFLG1
					PDBC	I	IPFLG1
					PRINT	I	IPFLG1
					TRTOSZ	O	IPFLG1
IPFLG2		IPFLG2≠0 supresses print-out of orbital parameters.	/GLOBAL/(	70)	PRINT	I	IPFLG2
					TRTOSZ	O	IPFLG2
IPFLG3		IPFLG3≠0 supresses print-out of impact data.	/GLOBAL/(	71)	OUT	I	IPFLG3
					PRINT	I	IPFLG3
					TRTOSZ	O	IPFLG3
IPFLG4		IPFLG4≠0 supresses print-out of inertial Cartesian coordinates.	/GLOBAL/(	72)	PRINT	I	IPFLG4
INEQFL		A 20 word array that contains the code number of the state variables inequality constraint that applies on each subarc. A zero entry indicates that no SVIC applies.	/GLOBAL/(	73)	PROPB	I	INEQFL
					PROPIN	M	INEQFL
					STP3	M	INEQFL
JP3		Description not input	/GLOBAL/(	92)	ANLATM	I	JP3
					MODELA	M	INQF
					MODELB	O	INQF
					PAT63	I	JP3
					PROPIN	M	INQF
ITPSO		A non zero input value indicates to the steepest descent module that the initial steering angle profiles are stored on logical unit 11.	/GLOBAL/(	93)	SDINP	I	ITPSO
KSOL		An internal flag that has the same significance as ITPSO.	/GLOBAL/(	94)	FNTG	I	KSOL
					PADS1	O	KSOL
					SDINP	M	KSOL

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
INARK		Logical unit on which initial and converged arcs are stored. INARK = 11.	/GLOBAL/(	95)			
KGLOBL		A seven word array not used.	/GLOBAL/(	96)	PADS1	I	ICONVR

BLØCK
LASTAB

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
LASTAB		Last word index	/LASTAB/(	1)	BLICO GEINP SOINP SPLICO TABIN	I I I I O	LASTAB LSTWD LSTWD LASTAB LASTAB
IPCT		Input Part case number	/LASTAB/(	2)	TABIN	M	IPCT
IPPNTR		Pointer array	/LASTAB/(	35)	PADS1 TABIN	D M	IPPNTR IPPNTR
NTAPE		Input tape unit = 1	/LASTAB/(	36)	PADS1 TABIN	D I	NTAPE NTAPE
MAXPCT		Maximum number of part cases = 33	/LASTAB/(	37)	PADS1 TABIN	D I	MAXPCT MAXPCT
BLANK		Preset	/LASTAB/(	38)	PADS1 TABIN	D I	BLANK BLANK

30 OCT 72 G.01-46

BLØCK
ØRBIT

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
VI	V_I	Inertial velocity (FT/SEC)	/ORBIT /(	1)	OUT PDBC PDBC	I I M	VI ORBPRM VI
ORBPRM		Array of intertial and orbital boundary conditions	/ORBIT /(	1)			
GAMI	γ_I	Inertial flight path angles (RAD)	/ORBIT /(	2)	OUT PDBC	I O	GAMI GAMI
PSII	ψ_I	Inertial azimuth (RAD)	/ORBIT /(	3)	OUT PDBC	I M	PSII PSII
XMUI	μ_I	Inertial longitude (RAD)	/ORBIT /(	4)	OUT PDBC	I M	XMUI XMUI
P	p_r	Semi-latus rectum (FT)	/ORBIT /(	5)	OUT PDBC	I M	P P
ECC	e	Orbital eccentricity	/ORBIT /(	6)	OUT PDBC	I M	ECC ECC
AINCL	i	Orbital inclination (RAD)	/ORBIT /(	7)	OUT PDBC	I M	AINCL AINCL
ARGP	ω_p	Orbital argument of perigee (RAD)	/ORBIT /(	8)	OUT PDBC	I M	ARGP ARGP
AXCNOD	Ω	Longitude of ascending node (RAD)	/ORBIT /(	9)	OUT PDBC	I M	ASCNOD ASCNOD
SMIMAJ	a_s	Semi-major axis (FT)	/ORBIT /(	10)	OUT PDBC	I M	SMIMAJ SMIMAJ
APOGEE	R_a	Apogee radius (FT)	/ORBIT /(	11)	OUT PDBC	I O	APOGEE APOGEE
PERGEE	R_p	Perigee radius (FT)	/ORBIT /(	12)	OUT PDBC	I O	PERGEE PERGEE
ANOMLY	ζ	True anomaly (RAD)	/ORBIT /(	13)	OUT PDBC	I M	ANOMLY ANOMLY
CAPX	X	Asymptote parameter T	/ORBIT /(	14)	PDBC	O	CAPX
CAPY	Y	Asymptote parameter (FT)	/ORBIT /(	15)	PDBC	O	CAPY
ASYMP	θ	Outgoing asymptote (RAD)	/ORBIT /(	16)	PDBC	M	ASYMP
ENERGY	E	Energy	/ORBIT /(	17)	OUT PDBC	I O	ENERGY ENERGY
HMNTM	H	Momentum	/ORBIT /(	18)	OUT PDBC	I M	HMNTM HMNTM
DVIDV		Partial derivative of boundary condition	/ORBIT /(	19)	PDBC PDBC	M I	DVIDV PPO
PPO		Matrix of boundary condition partial derivatives	/ORBIT /(	19)			
DVIDG		Partial derivative of boundary condition	/ORBIT /(	20)	PDBC	M	DVIDG
DVIDH		Partial derivative of boundary condition	/ORBIT /(	21)	PDBC	M	DVIDH
DVIDM		Partial derivative of boundary condition	/ORBIT /(	22)	PDBC	O	DVIDM
DVIDPS		Partial derivative of boundary condition	/ORBIT /(	23)	PDBC	M	DVIDPS
DVIDRO		Partial derivative of boundary condition	/ORBIT /(	24)	PDBC	M	DVIDRO
DVIDMU		Partial derivative of boundary condition	/ORBIT /(	25)	PDBC	O	DVIDMU
DGIDV		Partial derivative of boundary condition	/ORBIT /(	26)	PDBC	M	DGIDV
DGIDG		Partial derivative of boundary condition	/ORBIT /(	27)	PDBC	M	DGIDG
DGIDH		Partial derivative of boundary condition	/ORBIT /(	28)	PDBC	M	DGIDH
DGIDM		Partial derivative of boundary condition	/ORBIT /(	29)	PDBC	O	DGIDM
DGIDPS		Partial derivative of boundary condition	/ORBIT /(	30)	PDBC	M	DGIDPS
DGIDRO		Partial derivative of boundary condition	/ORBIT /(	31)	PDBC	M	DGIDRO
DGIDMU		Partial derivative of boundary condition	/ORBIT /(	32)	PDBC	O	DGIDMU
DPIDV		Partial derivative of boundary condition	/ORBIT /(	33)	PDBC	M	DPIDV
DPIDG		Partial derivative of boundary condition	/ORBIT /(	34)	PDBC	M	DPIDG

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DPIDH		Partial derivative of boundary condition	/ORBIT /(	35)	PDBC	M	DPIDH
DPIDM		Partial derivative of boundary condition	/ORBIT /(	36)	PDBC	O	DPIDM
DPIDPS		Partial derivative of boundary condition	/ORBIT /(	37)	PDBC	M	DPIDPS
DPIDRO		Partial derivative of boundary condition	/ORBIT /(	38)	PDBC	M	DPIDRO
DPIDMU		Partial derivative of boundary condition	/ORBIT /(	39)	PDBC	O	DPIDMU
DMIDV		Partial derivative of boundary condition	/ORBIT /(	40)	PDBC	O	DMIDV
DMIDG		Partial derivative of boundary condition	/ORBIT /(	41)	PDBC	O	DMIDG
DMIDH		Partial derivative of boundary condition	/ORBIT /(	42)	PDBC	O	DMIDH
DMIDM		Partial derivative of boundary condition	/ORBIT /(	43)	PDBC	O	DMIDM
DMIDPS		Partial derivative of boundary condition	/ORBIT /(	44)	PDBC	O	DMIDPS
DMIDRO		Partial derivative of boundary condition	/ORBIT /(	45)	PDBC	O	DMIDRO
DMIDMU		Partial derivative of boundary condition	/ORBIT /(	46)	PDBC	O	DMIDMU
DPDV		Partial derivative of boundary condition	/ORBIT /(	47)	PDBC	M	DPDV
DPDG		Partial derivative of boundary condition	/ORBIT /(	48)	PDBC	M	DPDG
DPDH		Partial derivative of boundary condition	/ORBIT /(	49)	PDBC	M	DPDH
DPDM		Partial derivative of boundary condition	/ORBIT /(	50)	PDBC	O	DPDM
DPDPS		Partial derivative of boundary condition	/ORBIT /(	51)	PDBC	M	DPDPS
DPDRO		Partial derivative of boundary condition	/ORBIT /(	52)	PDBC	M	DPDRO
DPDMU		Partial derivative of boundary condition	/ORBIT /(	53)	PDBC	O	DPDMU
DECDV		Partial-derivative of boundary condition	/ORBIT /(	54)	PDBC	M	DECDV
DECDG		Partial derivative of boundary condition	/ORBIT /(	55)	PDBC	M	DECDG
DECDH		Partial derivative of boundary condition	/ORBIT /(	56)	PDBC	M	DECDH
DECDM		Partial derivative of boundary condition	/ORBIT /(	57)	PDBC	O	DECDM
DECDPS		Partial derivative of boundary condition	/ORBIT /(	58)	PDBC	M	DECDPS
DECDRO		Partial derivative of boundary condition	/ORBIT /(	59)	PDBC	M	DECDRO
DECDMU		Partial derivative of boundary condition	/ORBIT /(	60)	PDBC	O	DECDMU
DIDV		Partial derivative of boundary condition	/ORBIT /(	61)	PDBC	M	DIDV
DIDG		Partial derivative of boundary condition	/ORBIT /(	62)	PDBC	M	DIDG
DIDH		Partial derivative of boundary condition	/ORBIT /(	63)	PDBC	M	DIDH
DIDM		Partial derivative of boundary condition	/ORBIT /(	64)	PDBC	O	DIDM
DIDPS		Partial derivative of boundary condition	/ORBIT /(	65)	PDBC	M	DIDPS
DIDRO		Partial derivative of boundary condition	/ORBIT /(	66)	PDBC	M	DIDRO
DIDMU		Partial derivative of boundary condition	/ORBIT /(	67)	PDBC	O	DIDMU
DBEDV		Partial derivative of boundary condition	/ORBIT /(	68)	PDBC	O	DBEDV
DBEDG		Partial derivative of boundary condition	/ORBIT /(	69)	PDBC	O	DBEDG
DBEDH		Partial derivative of boundary condition	/ORBIT /(	70)	PDBC	O	DBEDH
DBEDM		Partial derivative of boundary condition	/ORBIT /(	71)	PDBC	O	DBEDM
DBEDPS		Partial derivative of boundary condition	/ORBIT /(	72)	PDBC	O	DBEDPS
DBEDRO		Partial derivative of boundary condition	/ORBIT /(	73)	PDBC	O	DBEDRO
DBEDMU		Partial derivative of boundary condition	/ORBIT /(	74)	PDBC	O	DBEDMU
DNODV		Partial derivative of boundary condition	/ORBIT /(	75)	PDBC	O	DNODV
DNODG		Partial derivative of boundary condition	/ORBIT /(	76)	PDBC	O	DNODG
DNODH		Partial derivative of boundary condition	/ORBIT /(	77)	PDBC	O	DNODH
DNODM		Partial derivative of boundary condition	/ORBIT /(	78)	PDBC	O	DNODM
DNODPS		Partial derivative of boundary condition	/ORBIT /(	79)	PDBC	O	DNODPS
DNODRO		Partial derivative of boundary condition	/ORBIT /(	80)	PDBC	O	DNODRO
DNODMU		Partial derivative of boundary condition	/ORBIT /(	81)	PDBC	O	DNODMU
DSMDV		Partial derivative of boundary condition	/ORBIT /(	82)	PDBC	M	DSMDV
DSMDG		Partial derivative of boundary condition	/ORBIT /(	83)	PDBC	M	DSMDG

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DSMDH		Partial derivative of boundary condition	/ORBIT /(	84)	PDBC	M	DSMDH
DSMDM		Partial derivative of boundary condition	/ORBIT /(	85)	PDBC	O	DSMDM
DSMDPS		Partial derivative of boundary condition	/ORBIT /(	86)	PDBC	M	DSMDPS
DSMDRO		Partial derivative of boundary condition	/ORBIT /(	87)	PDBC	M	DSMDRO
DSMDMU		Partial derivative of boundary condition	/ORBIT /(	88)	PDBC	O	DSMDMU
DAPDV		Partial derivative of boundary condition	/ORBIT /(	89)	PDBC	O	DAPDV
DAPDG		Partial derivative of boundary condition	/ORBIT /(	90)	PDBC	O	DAPDG
DAPDH		Partial derivative of boundary condition	/ORBIT /(	91)	PDBC	O	DAPDH
DAPDM		Partial derivative of boundary condition	/ORBIT /(	92)	PDBC	O	DAPDM
DAPDPS		Partial derivative of boundary condition	/ORBIT /(	93)	PDBC	O	DAPDPS
DAPDRO		Partial derivative of boundary condition	/ORBIT /(	94)	PDBC	O	DAPDRO
DAPDMU		Partial derivative of boundary condition	/ORBIT /(	95)	PDBC	O	DAPDMU
DPEDV		Partial derivative of boundary condition	/ORBIT /(	96)	PDBC	O	DPEDV
DPEDG		Partial derivative of boundary condition	/ORBIT /(	97)	PDBC	O	DPEDG
DPEDH		Partial derivative of boundary condition	/ORBIT /(	98)	PDBC	O	DPEDH
DPEDM		Partial derivative of boundary condition	/ORBIT /(	99)	PDBC	O	DPEDM
DPEDPS		Partial derivative of boundary condition	/ORBIT /(	100)	PDBC	O	DPEDPS
DPEDRO		Partial derivative of boundary condition	/ORBIT /(	101)	PDBC	O	DPEDRO
DPEDMU		Partial derivative of boundary condition	/ORBIT /(	102)	PDBC	O	DPEDMU
DANDV		Partial derivative of boundary condition	/ORBIT /(	103)	PDBC	O	DANDV
DANDG		Partial derivative of boundary condition	/ORBIT /(	104)	PDBC	O	DANDG
DANDH		Partial derivative of boundary condition	/ORBIT /(	105)	PDBC	O	DANDH
DANDM		Partial derivative of boundary condition	/ORBIT /(	106)	PDBC	O	DANDM
DANDPS		Partial derivative of boundary condition	/ORBIT /(	107)	PDBC	O	DANDPS
DANDRO		Partial derivative of boundary condition	/ORBIT /(	108)	PDBC	O	DANDRO
DANDMU		Partial derivative of boundary condition	/ORBIT /(	109)	PDBC	O	DANDMU
DCXDV		Partial derivative of boundary condition	/ORBIT /(	110)	PDBC	O	DCXDV
DCXDG		Partial derivative of boundary condition	/ORBIT /(	111)	PDBC	O	DCXDG
DCXDH		Partial derivative of boundary condition	/ORBIT /(	112)	PDBC	O	DCXDH
DCXDM		Partial derivative of boundary condition	/ORBIT /(	113)	PDBC	O	DCXDM
DCXDPS		Partial derivative of boundary condition	/ORBIT /(	114)	PDBC	O	DCXDPS
DCXDRO		Partial derivative of boundary condition	/ORBIT /(	115)	PDBC	O	DCXDRO
DCXDMU		Partial derivative of boundary condition	/ORBIT /(	116)	PDBC	O	DCXDMU
DCYDV		Partial derivative of boundary condition	/ORBIT /(	117)	PDBC	O	DCYDV
DCYDG		Partial derivative of boundary condition	/ORBIT /(	118)	PDBC	O	DCYDG
DCYDH		Partial derivative of boundary condition	/ORBIT /(	119)	PDBC	O	DCYDH
DCYDM		Partial derivative of boundary condition	/ORBIT /(	120)	PDBC	O	DCYDM
DCYDPS		Partial derivative of boundary condition	/ORBIT /(	121)	PDBC	O	DCYDPS
DCYDRO		Partial derivative of boundary condition	/ORBIT /(	122)	PDBC	O	DCYDRO
DCYDMU		Partial derivative of boundary condition	/ORBIT /(	123)	PDBC	O	DCYDMU
DASDV		Partial derivative of boundary condition	/ORBIT /(	124)	PDBC	O	DASDV
DASDG		Partial derivative of boundary condition	/ORBIT /(	125)	PDBC	O	DASDG
DASDH		Partial derivative of boundary condition	/ORBIT /(	126)	PDBC	O	DASDH
DASDM		Partial derivative of boundary condition	/ORBIT /(	127)	PDBC	O	DASDM
DASDPS		Partial derivative of boundary condition	/ORBIT /(	128)	PDBC	O	DASDPS
DASDRO		Partial derivative of boundary condition	/ORBIT /(	129)	PDBC	O	DASDRO
DASDMU		Partial derivative of boundary condition	/ORBIT /(	130)	PDBC	O	DASDMU
DENDV		Partial derivative of boundary condition	/ORBIT /(	131)	PDBC	O	DENDV
DENDG		Partial derivative of boundary condition	/ORBIT /(	132)	PDBC	O	DENDG

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DENDH		Partial derivative of boundary condition		/ORBIT /(	133)	PDBC	O	DENDH
DENDM		Partial derivative of boundary condition		/ORBIT /(	134)	PDBC	O	DENDM
DENDPS		Partial derivative of boundary condition		/ORBIT /(	135)	PDBC	O	DENDPS
DENDRO		Partial derivative of boundary condition		/ORBIT /(	136)	PDBC	O	DENDRO
DENDMU		Partial derivative of boundary condition		/ORBIT /(	137)	PDBC	O	DENDMU
DMODV		Partial derivative of boundary condition		/ORBIT /(	138)	PDBC	O	DMODV
DMODG		Partial derivative of boundary condition		/ORBIT /(	139)	PDBC	O	DMODG
DMODH		Partial derivative of boundary condition		/ORBIT /(	140)	PDBC	O	DMODH
DMODM		Partial derivative of boundary condition		/ORBIT /(	141)	PDBC	O	DMODM
DMODPS		Partial derivative of boundary condition		/ORBIT /(	142)	PDBC	O	DMODPS
DMODRO		Partial derivative of boundary condition		/ORBIT /(	143)	PDBC	O	DMODRO
DMODMU		Partial derivative of boundary condition		/ORBIT /(	144)	PDBC	O	DMODMU
YMXRF	ρ_r	Reference longitude	(RAD)	/ORBIT /(	145)	PDBC	I	YMXRF
						REU3	O	YMXRF
SNXLMR	$\sin(\rho-\rho_r)$	Sine of reference latitude		/ORBIT /(	146)	PDBC	I	SNXLMR
						REU3	O	SNXLMR
CSXLMR	$\cos(\rho-\rho_r)$	Cosine of refernece latitude		/ORBIT /(	147)	PDBC	I	CSXLMR
						REU3	O	CSXLMR
SDOWN	S_D	Down range	(FT)	/ORBIT /(	148)	OUT	I	SDOWN
						PDBC	O	SDOWN
SCROSS	S_C	Cross range	(FT)	/ORBIT /(	149)	OUT	I	SCROSS
						PDBC	O	SCROSS
TD	θ_D	Down range angle	(RAD)	/ORBIT /(	150)	PDBC	M	TD
TC	θ_C	Cross range angle	(RAD)	/ORBIT /(	151)	PDBC	M	TC
SNPSR	$\sin(\psi_r)$	Sine of reference azimuth		/ORBIT /(	152)	PDBC	I	SNPSR
						REU3	O	SNPSR
CSPSR	$\cos(\psi_r)$	Cosine of reference azimuth		/ORBIT /(	153)	PDBC	I	CSPSR
						REU3	O	CSPSR
SNGI	$\sin(\gamma_I)$	Sine of inertial flight path angle		/ORBIT /(	154)	PDBC	M	SNGI
CSGI	$\cos(\gamma_I)$	Cosine of inertial flight path angle		/ORBIT /(	155)	PDBC	M	CSGI
SPSII	$\sin(\psi_I)$	Sin of inertial azimuth		/ORBIT /(	156)	PDBC	M	SPSII
CSPSII	$\cos(\psi_I)$	Cosine of inertial azimuth		/ORBIT /(	157)	PDBC	M	CPSII
STOT	S_T	Total range	(FT)	/ORBIT /(	158)	OUT	I	STOT
						PDBC	O	STOT
						TRTOSZ	I	STOT
CSI	$\cos(i)$	Cosine of inclination		/ORBIT /(	159)	PDBC	M	CSI
SNI	$\sin(i)$	Sine of inclination		/ORBIT /(	160)	PDBC	M	SNI
SNGNU	$\sin(\nu)$	See symbol		/ORBIT /(	161)	PDBC	M	SNGNU
CSANO	$\cos(\xi)$	See symbol		/ORBIT /(	162)	PDBC	M	CSANO
COSDMU	$\cos(\mu-\mu_r)$	See symbol		/ORBIT /(	163)	PDBC	M	COSDMU
SINDMU	$\sin(\mu-\mu_r)$	See symbol		/ORBIT /(	164)	PDBC	M	SINDMU
THT	θ_T	Total range angle	(FT)	/ORBIT /(	165)	PDBC	M	THT

BLØCK SIZING

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SW		A synthesis array (20) containing counters and sizing options	/SIZING/(	26)			
SV		A synthesis array (28) containing staging parameters and misc flags	/SIZING/(	46)	ENVPRM	M	SV
					FLYBKP	I	SV
					ITER8	I	SV
					RANGE	I	SV
					SIZEMR	M	SV
					SIZIN	I	SV
					SSSP	I	SV
					SUMOUT	I	SV
					TAMPAR	O	SV
					TAMPER	M	SV
					TRTOSZ	M	SV
					VEHDF	M	SV
					WTVOL	I	SV
SQ		A synthesis data array (37,5) that contains the flyback data and some injection quantities	/SIZING/(	74)	ENVPRM	M	SQ
					FLYBKP	M	SQ
					ISPRAT	I	SQ
					PDBC	I	SQ
					PRITVA	I	SQ
					RANGE	M	SQ
					REU3	O	SQ
					SIZE	O	SQ
					SIZEMR	M	SQ
					SIZIN	M	SQ
					STAU	I	SQ
					SUMOUT	M	SQ
					TAMPAR	O	SQ
					TAMPER	M	SQ
					THRUST	M	SQ
					TRTOSZ	M	SQ
					VEHDF	M	SQ
					WTVOL	M	SQ
SE		Array of synthesis iteration propulsion parameters	/SIZING/(	259)	FLYBKP	O	SE
					PRITVA	I	SE
					SIZEMR	I	SE
					SUMOUT	I	SE
					TAMPER	M	SE
					THRUST	I	SE
					VEHDF	M	SE
					WTSCH	I	SE
					WTVOL	M	SE
QLIM		Saved value of maximum dynamic pressure.	/SIZING/(	264)	ENVPRM	M	QLIM
TLAT		Description not input	/SIZING/(	270)	SUMOUT	I	TLAT
TLNG		Description not input	/SIZING/(	271)	SUMOUT	I	TLNG
WBO		Booster burnout weight (lb)	/SIZING/(	272)	GEINP	M	SIZ
					PAYLOD	O	WBO
					SIZE	I	DAT
					SIZOUT	I	WBO
					SIZ1	M	WBO
					SIZ2	M	WBO
					SIZ4	M	WBO
					TAMPAR	I	WBO
WLOO		Previous iteration value of booster liftoff weight (lb)	/SIZING/(	273)	SIZE	M	WLOO
DWEB		Sensitivity of booster stage weight to propellant weight (lb/lb)	/SIZING/(	274)	PAYLOD	I	DWEB
					SIZOUT	I	DWEB
					SIZ1	M	DWEB
					SIZ2	M	DWEB
					STAU	I	DWEB
					WTORP	M	DWEB
DWEO		Sensitivity of orbiter stage weight to propellant weight (lb/lb)	/SIZING/(	275)	PAYLOD	I	DWEO
					SIZOUT	I	DWEO
					SIZ1	M	DWEO
					SIZ2	M	DWEO
					SIZ4	M	DWEO
					WTORP	M	DWEO
TOLWT		Booster liftoff weight sizing tolerance (lb)	/SIZING/(	276)	SIZE	M	TOLWT

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
WPB		Booster propellant weight (lb)	/SIZING/(	277)	SIZOUT	I	WPB
					SIZ1	M	WPB
					SIZ2	M	WPB
					SIZ4	I	WPB
					TAMPAR	I	WPB
					WTORP	M	WPB
TWRAT2		Second st0ge thrust-to-weight ratio	/SIZING/(	278)	SIZOUT	I	TWRAT2
					SIZ1	O	TWRAT2
					SIZ2	O	TWRAT2
					SIZ3	O	TWRAT2
					SIZ4	O	TWRAT2
BK1		Value of constant weight in booster stage weight equation	/SIZING/(	279)	SIZ1	I	BK1
					SIZ2	I	BK1
					WTORP	I	BK1
BK2		Value of linear term coefficient in booster stage weight equation	/SIZING/(	280)	SIZ1	I	BK2
					SIZ2	I	BK2
					WTORP	I	BK2
BK3		Value of 1/3-power term coefficient in booster stage weight equation	/SIZING/(	281)	SIZ1	I	BK3
					SIZ2	I	BK3
					WTORP	I	BK3
BK4		Value of 2/3-power term coefficient in booster stage weight equation	/SIZING/(	282)	SIZ1	I	BK4
					SIZ2	I	BK4
					WTORP	I	BK4
ISIZE		Sizing option flag 1. Fixed wlo, maximize wpl 2. Fixed wpl, minimize wlo 3. Fixed orbiter, minimize wlo 4. Fixed booster, minimize wlo 5. Fixed (t/w)1.0. Maximize wpl 6. Fixed (t/w)1.0. Determine f	/SIZING/(	283)	SIZE	M	ISIZE
					SIZOUT	I	ISIZE
TRAFLG		Traffic control flag 0. Sizing loop not converged 1. Sizing loop converged 2. Error in sizing loop next case	/SIZING/(	284)	FRENCH	O	TRAFLG
					ITER8	O	TRAFLG
					PADS1	I	TRAFLG
					SIZE	M	TRAFLG
					SSSP	O	TRAFLG
					VEHOF	O	TRAFLG
TWRATO		Liftoff thrust-to-weight ratio	/SIZING/(	285)	SIZOUT	I	TWRATO
					SIZ1	O	TWRATO
					SIZ2	O	TWRATO
					SIZ4	O	TWRATO
					SIZ5	I	TWRATO
OK1		Same as bk1 except for orbiter	/SIZING/(	286)	SIZ1	I	OK1
					SIZ2	I	OK1
					SIZ4	I	OK1
					WTORP	I	OK1
OK2		Same as bk2 except for orbiter	/SIZING/(	287)	SIZ1	I	OK2
					SIZ2	I	OK2
					SIZ4	I	OK2
					WTORP	I	OK2
OK3		Same as bk3 except for orbiter	/SIZING/(	288)	SIZ1	I	OK3
					SIZ2	I	OK3
					SIZ4	I	OK3
					WTORP	I	OK3
OK4		Same as bk4 except for orbiter	/SIZING/(	289)	SIZ1	I	OK4
					SIZ2	I	OK4
					SIZ4	I	OK4
					WTORP	I	OK4
PRFLG		Sizing data print flag 1. Print header 2. Print identifier 3. Print data	/SIZING/(	290)	SIZE	M	PRFLG
					SIZOUT	M	PRFLG
IPASS		Sizing iteration counter	/SIZING/(	291)	GEINP	O	IPASS
					PADS1	M	IPASS
					PAYO2	I	IPASS
					SIZE	M	IPASS
					SIZIN	I	IPASS
					SSSP	M	IPASS

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IPSMAX		Maximum number of iterations	/SIZING/(	292)	SIZE	M	IPSMAX
					SSSP	I	IPSMAX
					VEHDF	I	HIPSMX
					VEHDF	O	IPSMAX
AEXIT		Booster engine exit area (ft**2)	/SIZING/(	293)	SIZ1	I	AEXIT
					SIZ2	I	AEXIT
					SIZ4	I	AEXIT
					SIZ5	I	AEXIT
					TAMPAR	I	AEXIT
TVACO		Orbiter vacuum thrust (lb)	/SIZING/(	294)	SIZOUT	I	TVACO
					SIZ1	I	TVACO
					SIZ2	I	TVACO
					SIZ3	I	TVACO
					SIZ4	I	TVACO
					TAMPAR	I	TVACO
NO		Number of orbiter engines	/SIZING/(	295)	SIZOUT	I	NO
					SIZ1	I	NO
					SIZ2	I	NO
					SIZ3	I	NO
					SIZ4	I	NO
					TAMPAR	I	NO
WFO		Orbiter burnout weight (lb)	/SIZING/(	296)	PAYLOD	M	WFO
					SIZOUT	I	WFO
					SIZ1	O	WFO
					SIZ2	O	WFO
					SIZ3	O	WFO
					SIZ4	M	WFO
					TAMPAR	I	WFO
					TRTOSZ	O	WFO
IDVEL		Total ideal velocity required to orbit (fps)	/SIZING/(	297)	SIZ1	I	IDVEL
					SIZ2	I	IDVEL
					SIZ3	I	IDVEL
					SIZ4	I	IDVEL
					TRTOSZ	O	IDVEL
ISPO		Orbiter vacuum specific impulse sec	/SIZING/(	298)	SIZE	M	ISPO
					SIZOUT	I	ISPO
					SIZ1	I	ISPO
					SIZ2	I	ISPO
					SIZ3	I	ISPO
					SIZ4	I	ISPO
					TAMPAR	I	ISPO
ISPB		Booster vacuum specific impulse sec	/SIZING/(	299)	SIZE	M	ISPB
					SIZOUT	I	ISPB
					SIZ1	I	ISPB
					SIZ2	I	ISPB
					SIZ3	I	ISPB
					SIZ4	I	ISPB
					TAMPAR	I	ISPB
XPL		Payload weight (lb)	/SIZING/(	300)	SIZOUT	I	XPL
					SIZ1	O	XPL
					SIZ2	I	XPL
					SIZ3	I	XPL
					SIZ4	I	XPL
					TAMPAR	I	XPL
TVACB		Booster vacuum thrust per engine lb	/SIZING/(	301)	SIZOUT	I	TVACB
					SIZ1	I	TVACB
					SIZ2	I	TVACB
					SIZ4	I	TVACB
					SIZ5	I	TVACB
					TAMPAR	I	TVACB
NNB		Number of booster engines	/SIZING/(	302)	SIZOUT	I	NNB
					SIZ1	I	NNB
					SIZ2	I	NNB
					SIZ4	I	NNB
					SIZ5	I	NNB
					TAMPAR	I	NNB

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
WEO		Orbiter stage weight (lb)	/SIZING/(	303)	PAYLOD	I	WEO
					SIZOUT	I	WEO
					SIZ1	M	WEO
					SIZ2	M	WEO
					SIZ3	I	WEO
					SIZ4	M	WEO
					TAMPAR	I	WEO
					WTDRP	M	WEO
WEB		Booster stage weight (lb)	/SIZING/(	304)	PAYLOD	I	WEB
					SIZOUT	I	WEB
					SIZ1	M	WEB
					SIZ2	M	WEB
					SIZ4	I	WEB
					TAMPAR	I	WEB
					WTDRP	M	WEB
WO		Initial orbiter weight (lb)	/SIZING/(	305)	PAYLOD	O	WO
					SIZOUT	I	WO
					SIZ1	M	WO
					SIZ2	M	WO
					SIZ3	M	WO
					SIZ4	M	WO
					TAMPAR	I	WO
WLO		Booster liftoff weight (lb)	/SIZING/(	306)	PAYLOD	O	WLO
					SIZE	I	WLO
					SIZOUT	I	WLO
					SIZ1	I	WLO
					SIZ2	M	WLO
					SIZ4	M	WLO
					SIZ5	O	WLO
					TAMPAR	I	WLO
DVO		Orbiter ideal velocity (fps)	/SIZING/(	307)	SIZOUT	I	DVO
					SIZ1	O	DVO
					SIZ2	O	DVO
					SIZ3	M	DVO
					SIZ4	O	DVO
					TRTOSZ	O	DVO
DVB		Booster idel velocity (fps)	/SIZING/(	308)	SIZOUT	I	DVB
					SIZ1	M	DVB
					SIZ2	M	DVB
					SIZ3	O	DVB
					SIZ4	M	DVB
MUB		Booster mass ratio or velocity	/SIZING/(	309)	SIZE	M	MUB
					SIZ1	I	MUB
					SIZ2	I	MUB
					SIZ3	M	MUB
					SIZ4	M	MUB
MUO		Orbiter mass ratio	/SIZING/(	310)	SIZE	M	MUO
					SIZ1	M	MUO
					SIZ2	M	MUO
					SIZ4	M	MUO
VSTG		Booster staging velocity (fps)	/SIZING/(	311)	SIZE	I	VSTG
					TRTOSZ	M	VSTG
WPO		Orbiter propellant weight (lb)	/SIZING/(	312)	SIZOUT	I	WPO
					SIZ1	M	WPO
					SIZ2	M	WPO
					SIZ3	I	WPO
					SIZ4	M	WPO
					TAMPAR	I	WPO
					WTDRP	M	WPO
JTYP		Sizing. Flag.	/SIZING/(	313)	FNTG	I	JTYP
					GEINP	O	JTYP
					MODELA	I	JTYP
					PADSI	I	JTYP
					PROPIN	I	JTYP
					SIZIN	I	JTYP
					TRTOSZ	I	JTYP
BECO		Booster cut-off arc	/SIZING/(	314)	SIZE	I	BECO
					VEHDF	I	BECO

8 NOV 72 G.01-46

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
BSTG		Booster staging arc	/SIZING/(	315)	SIZE	I	BSTG
					VEHDF	I	BSTG
ORBI		Orbiter ignition arc	/SIZING/(	316)	REU3	I	ORBI
					SIZE	I	ORBI
					VEHDF	I	ORBI
ITNBW		Booster empty weght curve no.	/SIZING/(	317)	SIZE	O	ITNBW
					SIZ1	I	ITNBW
					SIZ2	I	ITNBW
					WTORP	I	ITNBW
ITNOW		Orbiter empty weight curve no.	/SIZING/(	318)	SIZE	O	ITNOW
					SIZ1	I	ITNOW
					SIZ2	I	ITNOW
					SIZ4	I	ITNOW
					WTORP	I	ITNOW
SVDPSQ		Saved control matric	/SIZING/(	319)	PAYO2	I	SVDPSQ
					TRTOSZ	O	SVDPSQ
SVDCON		Saved payoff improvement	/SIZING/(	320)	TRTOSZ	O	SVDCON
IHUNT		Number of iterations for parameter hunt	/SIZING/(	321)	GEINP	O	IHUNT
					OPWELL	I	ELIM
					SSSP	I	IHUNT
IOPSTG		Description not input	/SIZING/(	322)	GEINP	O	IOPSTG
					SIZIN	O	IOPSTG
					SSSP	I	IOPSTG
SOCO		Solid engine cut-off arc	/SIZING/(	326)	VEHDF	I	SOCO
SOSP		Solid engine drop arc	/SIZING/(	327)	VEHDF	I	SOSP

BLØCK TABLE

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
LOCI		A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the initial value of the independent variable of the corresponding table. A zero entry indicates table not input.	/TABLE /(	1)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP THRUP	M M I I I I I O	LOCI X LOCI X LOCI X LOCI X
TABLE		A 2100 word array used for storing up to 50 spline fitted univariant tables.	/TABLE /(	1)			
X		A 650 word array that contains the independent variable entries of all of the 50 or less tables input.	/TABLE /(	51)			
LOCL		A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that indicates the last interval in which interpolation of the corresponding table occured.	/TABLE /(	701)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP	O M M I M I O	LOCL Y LOCL Y LOCL Y Y
Y		A 650 word array that contains the dependent variable entries of all of the 50 or less tables input.	/TABLE /(	751)			
LOCF		A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the last value of the independent variable of the corresponding table.	/TABLE /(	1401)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP	M M I I I I O	LOCF Z LOCF Z LOCF Z Z
Z		The mesh point second derivatives of the cubic spline functions of all of the 50 or less tables input.	/TABLE /(	1451)			

SUBRØUTINE
CØØRDS

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ER	E_R	I	Earth radius.	(FT)	/GLOBAL/(	2)	COORDS	I	ER
							CRASH	I	REA
							EQUA3	I	ER
							GEINP	I	ER
							PADS1	I	ER
							PQBC	I	ER
							SOMG	I	ER
							TRTOSZ	I	ER

```
      SUBROUTINE COORDS(     COSRHO,SINRHO)                           COORDS
C                                                                     COORDS
C         SUBROUTINE TO CALCULATE INERTIAL POSITION, VELOCITY, AND    COORDS
C.        EULER ANGLES FROM RELATIVE STATES.                          COORDS
C                                                                     COORDS
      DIMENSION  A(9) , B(9), C(9), DIRCOS(9) ,EULER(3)               COORDS
      COMMON/GLOBAL/ GR,ER                                            COORDS
      COMMON/PRINT/ AP(100)                                           COORDS
C                                                                     COORDS
C         DATA INITIATION                                             APRL6
C                                                                     COORDS
      RAD= 1./57.29577951                                             APRL6
      PI= 3.141592653                                                 APRL6
      COSA   = COS(AP(29)*RAD)                                        COORDS
      SINA   = SIN(AP(29)*RAD)                                        COORDS
      COSBNK = COS(AP(30)*RAD)                                        COORDS
      SINBNK = SIN(AP(30)*RAD)                                        COORDS
      COSGAM = COS(AP(4)*RAD)                                         COORDS
      SINGAM = SIN(AP(4)*RAD)                                         COORDS
      SINPSI = SIN(AP(5)*RAD)                                         COORDS
      COSPSI = COS(AP(5)*RAD)                                         COORDS
      COSMUI = COS(AP(13)* RAD)                                       COORDS
      SINMUI = SIN(AP(13)* RAD)                                       COORDS
      R      =  ER + AP(2)                                            COORDS
C                                                                     APRL6
C         POSITION TRANSFORMATION                                     APRL6
C                                                                     APRL6
      AP(86) = R*  COSRHO * COSMUI                                    COORDS
      AP(87) = R*  COSRHO * SINMUI                                    COORDS
      AP(88) = R*  SINRHO                                             COORDS
C                                                                     COORDS
C         VELOCITY TRANSFORMATION                                     COORDS
C                                                                     COORDS
      RHODOT = AP(3)/R * COSGAM* COSPSI                               COORDS
      UMIDOT = AP(3)/R * COSGAM* SINPSI/ COSRHO                       COORDS
      RDOT   = AP(3)    * SINGAM                                      COORDS
      AP(89) = RDOT* AP(86)/R - RHODOT* AP(88)* COSMUI                COORDS
     1         - UMIDOT * AP(87)                                      COORDS
      AP(90) = RDOT* AP(87)/R - RHODOT* AP(88)* SINMUI                COORDS
     1         +UMIDOT * AP(86)                                       COORDS
      AP(91) = RDOT*SINRHO + R*RHODOT*COSRHO                          COORDS
C                                                                     COORDS
C         EULER  ANGLES                                               COORDS
C                                                                     COORDS
      DO 1   I=1,9                                                    COORDS
      A(I) =0.                                                        COORDS
      B(I) =0.                                                        COORDS
    1 C(I) =0.                                                        COORDS
      A(1) = 1.                                                       COORDS
      A(5) = 1.                                                       COORDS
      A(9) = 1.                                                       COORDS
      B(1) = COSA                                                     COORDS
      B(3)= SINA                                                      APRL6
      B(5) = 1.                                                       COORDS
      B(7) = -SINA                                                    APRL6
      B(9) = COSA                                                     COORDS
      CALL PRODUT(A,B,C)                                              COORDS
      DO 2 I=1,9                                                      COORDS
      B(I) = C(I)                                                     COORDS
    2 A(I) = 0.                                                       COORDS
      A(1) = 1.                                                       COORDS
      A(5) = COSBNK                                                   COORDS
      A(6) =-SINBNK                                                   COORDS
      A(8) = SINBNK                                                   COORDS
      A(9) = COSBNK                                                   COORDS
      CALL PRODUT (A,B,C)                                             COORDS
      DO 3 I=1,9                                                      COORDS
      B(I) = C(I)                                                     COORDS
    3 A(I) =0.                                                        COORDS
      A(1) = COSGAM                                                   COORDS
      A(3)= SINGAM                                                    APRL6
      A(5) = 1.                                                       COORDS
      A(7) = -SINGAM                                                  APRL6
      A(9) = COSGAM                                                   COORDS
      CALL  PRODUT (A,B,C)                                            COORDS
```

20 OCT 72 G.01-46

```
      DO 4  I=1,9                                                       COORDS
      B(I) =C(I)                                                        COORDS
    4 A(I) =0.                                                          COORDS
      A(1) = -COSPSI                                                    APRL6
      A(2) =-SINPSI                                                     COORDS
      A(4) = SINPSI                                                     COORDS
      A(5) = -COSPSI                                                    APRL6
      A(9) = 1.                                                         COORDS
      CALL PRODUT(A,B,DIRCOS)                                           COORDS
C                                                                       COORDS
C          RELATIVE EULER ANGLES                                        COORDS
C                                                                       COORDS
      CALL DCTOE (DIRCOS,EULER,1)                                       COORDS
      AP(52) = EULER(1)/RAD                                             COORDS
      AP(53) = EULER(2)/RAD                                             COORDS
      AP(54) = EULER(3)/RAD                                             COORDS
      DO 5 I=1,9                                                        COORDS
    5 B(I) = 0.                                                         COORDS
      B(1) = COSMUI                                                     COORDS
      B(2) = SINMUI                                                     APRL6
      B(4) = -SINMUI                                                    APRL6
      B(5) = COSMUI                                                     COORDS
      B(9) = 1.                                                         COORDS
      CALL PRODUT(DIRCOS,B,C)                                           COORDS
C                                                                       COORDS
C          INERTIAL EULER ANGLES                                        COORDS
C                                                                       COORDS
      CALL DCTOE(C,EULER,1)                                             COORDS
      AP(59) = EULER(1)/RAD                                             COORDS
      AP(60) = EULER(2)/RAD                                             COORDS
      AP(61) = EULER(3)/RAD                                             COORDS
      RETURN                                                            COORDS
      END                                                               COORDS
```

20 OCT 72 G.01-46

SUBRØUTINE CRASH

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
GM	GM	I	Product of Newton's universal gravitational constant and the mass of the earth. (FT^3/SEC^2)	/GLOBAL/(	67)	CRASH	I	GM
						OUT	I	GM
						PADS1	O	GM
						PDBC	I	GM
OMEGA	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
						CRASH	I	OMEGA
						DER3A	I	OMGZ
						EQUA3	I	OMGZ
						GEINP	I	OMGZ
						MODELA	I	OMGZ
						MODELB	I	OMGZ
						PDBC	I	OMGZ
						PDY3A	I	OMGZ
						SDINP	I	OMGZ
						TOPM	I	OMGZ
REM	E_R	I	Earth radius. (FT)	/GLOBAL/(	2)	COORDS	I	ER
						CRASH	I	REM
						EQUA3	I	ER
						GEINP	I	ER
						PADS1	I	ER
						PDBC	I	ER
						SOMG	I	ER
						TRTOSZ	I	ER
RHOO	ρ_r	M	Reference latitude. (DEG)	/GLOBAL/(	4)	CRASH	M	RHOO
						GEINP	I	XLAMRF
						REU3	I	XLAMRF
						SDINP	M	XLAMRF
SQRT		F	Square root function	/SQRT /($	)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTOE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAYD2	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
UMUO	μ_r	M	Reference longitude. (DEG)	/GLOBAL/(	5)	CRASH	M	UMUO
						GEINP	I	YMURF
						REU3	I	YMURF
						SDINP	M	YMURF

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
.UN06.		0	File of all output data	/.UN06./(	)	BLICO	0	.UN06.
						BNDRYC	0	.UN06.
						CRASH	0	.UN06.
						FRENCH	0	.UN06.
						FXDAT	0	.UN06.
						GEINP	0	.UN06.
						HUNT	0	.UN06.
						INEDIT	0	.UN06.
						ITERB	0	.UN06.
						MODELA	0	.UN06.
						MOMJ	0	.UN06.
						MPSI	0	.UN06.
						OUT	0	.UN06.
						PAY02	0	.UN06.
						PRINT	0	.UN06.
						PRINTV	0	.UN06.
						PRINTW	0	.UN06.
						PRITED	0	.UN06.
						PRITVA	0	.UN06.
						PROPIN	0	.UN06.
						PROTHR	0	.UN06.
						PRWTSM	0	.UN06.
						RANGE	0	.UN06.
						S	0	.UN06.
						SDINP	0	.UN06.
						SIZE	0	.UN06.
						SIZIN	0	.UN06.
						SIZOUT	0	.UN06.
						SOLVE	0	.UN06.
						SPLICO	0	.UN06.
						SPLIZ	0	.UN06.
						SPLYNE	0	.UN06.
						SSSP	0	.UN06.
						STAU	0	.UN06.
						STPIT	0	.UN06.
						SUMOUT	0	.UN06.
						TABIN	0	.UN06.
						TEST	0	.UN06.
						VEHDF	0	.UN06.
						WTSCH	0	.UN06.
						WTVOL	0	.UN06.

```
      SUBROUTINE CRASH                                                  CRASH
C                                                                       CRASH
C                               PADS  CRASH                             CRASH
C          THIS ROUTINE PERFORMS AUXILIARY PRINT COMPUTATIONS FOR       CRASH
C          THE INSTANTEOUS IMPACT POINTS FOR AN ELLIPTICAL ORBIT        CRASH
C          ALL ANGLES IN RADIANS                                        CRASH
C          ALL DISTANCES IN FEET                                        CRASH
C          ALL TIMES IN SECONDS                                         CRASH
C                                                                       CRASH
      COMMON/PRINT/  AP(100)                                            CRASH
      COMMON/GLOBAL/                                                    GLOBAL
     *GR         ,ER          ,OMGZ       ,XLAMRF      ,YMURF      ,LUM GLOBAL
     *,JJOP(10) ,IFATAL       ,NARC        ,NBRAN      ,NFARC     ,ID(4)GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                              GLOBAL
     *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)    GLOBAL
     *,ITP50 ,KSOL       ,KGLOBL(6)                                     RETAP
      REAL  NUTA, NUBAR, NUIIP                                          CRASH
      EQUIVALENCE                                                       CRASH
     *        (TIME ,AP(1))  ,(RIA,AP(67))   ,(RIP,AP(68))     ,        CRASH
     *      (E,AP(63))       ,(V,AP(10))     ,(GAM1IP,AP(11)),          CRASH
     *        (GAM2IP,AP(12)),(A,AP(62))     ,(P,AP(73))       ,        CRASH
     *        (RHOO,XLAMRF)  ,(UMUO,YMURF)   ,(REM,ER)         ,        CRASH
     *        (OMEGA,OMGZ )  ,(UMU,AP(6))    ,(RHO,AP(7))               CRASH
C                                                                       CRASH
C          CALCULATION OF IIP FOR ELLIPTICAL ORBITS                     CRASH
C                                                                       CRASH
      FTNM   =  1./6076.10333                                           CRASH
      RAD= 1./57.29577951                                               APRL6
      PI= 3.141592653                                                   APRL6
      TOPI = 2. * PI                                                    CRASH
      A = A/FTNM                                                        CRASH
      P = P/FTNM                                                        CRASH
      RIA = RIA/FTNM                                                    CRASH
      RIP = RIP/FTNM                                                    CRASH
      RHOO   = RHOO* RAD                                                CRASH
      UMUO   = UMUO* RAD                                                CRASH
      GAM1IP = GAM1IP*RAD                                               CRASH
      GAM2IP = GAM2IP* RAD                                              CRASH
      UMU    = UMU* RAD                                                 CRASH
      RHO    = RHO* RAD                                                 CRASH
      SINRP   = SIN(RHO)                                                CRASH
      COSRP   = COS(RHO)                                                CRASH
C                                                                       CRASH
C          RADIUS                                                       CRASH
C                                                                       CRASH
      RI      = REM + AP(2)                                             CRASH
C                                                                       CRASH
C          TEST FOR INTERSECTION OF ORBIT WITH EARTH                    CRASH
C                                                                       CRASH
      IF(REM.LT.RIP) GO TO 1000                                         CRASH  1000
      IF(REM.GT.RIA) GO TO 1001                                         CRASH  1001
C                                                                       CRASH
C          TRUE ANOMALY                                                 CRASH
C                                                                       CRASH
      COSY = ((P/RI) - 1.)/E                                            CRASH
      NUTA = ACOS(COSY)                                                 CRASH
      SINY = SIN(NUTA)                                                  CRASH
C                                                                       CRASH
C          IMPACT TRUE ANOMALY (NUIIP)                                  CRASH
C                                                                       CRASH
      COSNUI=    (P/REM -1.)/E                                          CRASH
      NUIIP= - ACOS(COSNUI)                                             CRASH
      SINNUI = SIN (NUIIP)                                              CRASH
C                                                                       CRASH
C          ECCENTRIC ANOMALY                                            CRASH
C                                                                       CRASH
      COSEO = (E + COSY)/(1.+E*COSY)                                    CRASH
      SINEO = (SQRT(ABS(1.-E*E))*SINY)/(1.+E*COSY)                      CRASH
      EC  = ATAN2(SINEO,COSEO)                                          CRASH
C                                                                       CRASH
C          IMPACT VELOCITY (VIIP)                                       CRASH
C                                                                       CRASH
      VIIP =SQRT(V*V+(GM/RI )*2.* (RI-REM)/REM)                         CRASH
C                                                                       CRASH
C          IMPACT ELEVATION FLIGHT PATH ANGLE  GAM1IF                   CRASH
```

20 OCT 72 6.01-46

```
C                                                               CRASH
      COSG1F = (RI*V*COS(GAM1IP    ))/(REM*VIIP)                CRASH
      GAM1IF =-  ACOS(COSG1F)                                   CRASH
C                                                               CRASH
C         TRUE ANOMALY TO IMPACT (NUBAR- RAD)                   CRASH
C                                                               CRASH
      NUBAR = (NUIIP    -NUTA)                                  CRASH
      NUBAR =  AMOD(NUBAR+TOPI,TOPI)                            CRASH
      SINBB = SIN (NUBAR)                                       CRASH
      COSBB = COS(NUBAR)                                        CRASH
C                                                               CRASH
C         IMPACT GEOCENTRIC LATITUDE (RHOIIP)                   CRASH
C                                                               CRASH
      CAM2IP = COS(GAM2IP)                                      CRASH
      SAM2IP = SIN(GAM2IP)                                      CRASH
      SINRIP = COSRP *    CAM2IP     *SINBB +                   CRASH
     1          SINRP *COSBB                                    CRASH
      RHOIIP =     ASIN(SINRIP)                                 CRASH
      COSRIP = COS(RHOIIP)                                      CRASH
C                                                               CRASH
C         IMPACT AZIMUTH FLIGHT PATH ANGLE  (GAM2IF)            CRASH
C                                                               CRASH
      SING2F  = COSRP *    SAM2IP     /COSRIP                   CRASH
      COSG2F = (COSBB*SINRIP-SINRP)/(SINBB)*COSRIP              CRASH
      GAM2IF  =  ATAN2(SING2F,COSG2F)                           CRASH
      GAM2IF = AMOD(GAM2IF + TOPI,TOPI)                         CRASH
C                                                               CRASH
C         LONGITUDE FROM BURNOUT TO IMPACT - (UMUBAR)           CRASH
C                                                               CRASH
      SUMBAR = SINBB*SING2F/COSRIP                              CRASH
      CUMBAR =(COSBB-SINRP*SINRIP)/(COSRP*COSRIP)               CRASH
      UMUBAR = ATAN2(SUMBAR,CUMBAR)                             CRASH
C                                                               CRASH
C         SIN OF IIP ECCENTRIC ANOMALY  (SINEF)                 CRASH
C                                                               CRASH
      SINEF =(SQRT(ABS(1.-E*E ))*SINNUI    )/(1.+E *COSNUI)     CRASH
C                                                               CRASH
C        COSINE OF ECCENTRIC ANOMALY AT IMPACT                  CRASH
C                                                               CRASH
      COSEF = (E +COSNUI)/(1.+E* COSNUI)                        CRASH
C                                                               CRASH
C         ECCENTRIC  ANOMALY AT IMPACT-EIIP                     CRASH
C                                                               CRASH
      EIIP =  ATAN2(SINEF,COSEF)                                CRASH
C                                                               CRASH
C         INITIAL TIME SINCE  PERIGEE -TAUO                     CRASH
C                                                               CRASH
      X      =  A* SQRT(A/GM)                                   CRASH
      TAU = TOPI*X                                              CRASH
      TAUO=X*(EC     -E*SINEO)                                  CRASH
C                                                               CRASH
C         TIME SINCE PERIGEE AT IMPACT TAUIIP                   CRASH
C                                                               CRASH
      TAUIIP = X*(EIIP    -E*SINEF)                             CRASH
C                                                               CRASH
C         TIME TO IMPACT -TAUBAR                                CRASH
C                                                               CRASH
      TAUBAR = TAUIIP - TAUO                                    CRASH
      IF(TAUBAR.LT.0.0) TAUBAR= TAUBAR + TAU                    CRASH
C                                                               CRASH
C         IMPACT LONGITUDE - UMUIIP  (DEG)                      CRASH
C                                                               CRASH
      UMUIIP =(UMU - UMUBAR)+TAUBAR*OMEGA                       CRASH
      UMUIIP = AMOD( UMUIIP+ TOPI,TOPI)                         CRASH
C                                                               CRASH
C         IMPACT TIME -TIIP  (SEC)                              CRASH
C                                                               CRASH
      TIIP = TIME + TAUBAR                                      CRASH
C                                                               CRASH
C         RANGE TO IMPACT  SBAR (FT)                            CRASH
C                                                               CRASH
      SBAR =REM*NUBAR                                           CRASH
C                                                               CRASH
C         RANGE FROM LAUNCH SITE   SIIP (FT)                    CRASH
C                                                               CRASH
```

20 OCT 72 6.01-46

```
      X  = (UMUO - UMUIIP)                                              CRASH
      SINROL = SIN(RHOO)                                                CRASH
      COSROL = COS(RHOO)                                                CRASH
      SX = SIN(X)                                                       CRASH
      CX = COS(X)                                                       CRASH
      COSX = COSROL       *COSRIP*CX+SINROL*SINRIP                      CRASH
      COSX =  ACOS(COSX)                                                CRASH
      COSX = AMOD(COSX + PI,PI)                                         CRASH
      SIIP = REM*COSX                                                   CRASH
C                                                                       CRASH
C          AZIMUTH ANGLE FROM LAUNCH TO IMPACT POINT AZIIP (RAD)        CRASH
C                                                                       CRASH
      IF(X.EQ.0.) GO TO 1002                                            CRASH
      GO TO 1010                                                        CRASH
 1002 IF(RHOIIP.LT.RHOO) GO TO 1003                                     CRASH
 1012 AZIIP=0.0                                                         CRASH
      GO TO 1030                                                        CRASH
 1003 AZIIP = PI                                                        CRASH
      GO TO 1030                                                        CRASH
 1010 IF(X.EQ.PI   )  GO TO 1011                                        CRASH
      GO TO 1020                                                        CRASH
 1011 IF(RHOIIP.GE.-RHOO)  GO TO 1012                                   CRASH
      GO TO 1003                                                        CRASH
 1020 CONTINUE                                                          CRASH
      AZIIP = ACOS((SX/ABS(SX)/SIN(SIIP/REM))                           CRASH
     *      *(COSROL*SINRIP-COSRIP*SINROL*CX))                          CRASH
     *        -((SX/ABS(SX))-1.)*PI/2.                                  CRASH
 1030 CONTINUE                                                          CRASH
      AP(76)  = TIIP/60.                                                CRASH
      AP(77)  = SIIP*FTNM                                               CRASH
      AP(78)  = AZIIP/ RAD                                              CRASH
      AP(79)  = VIIP                                                    CRASH
      AP(80)  = GAM1IF/ RAD                                             CRASH
      AP(81)  = RHOIIP/ RAD                                             CRASH
      AP(82)  = TAUBAR/60.                                              CRASH
      AP(83)  = SBAR*FTNM                                               CRASH
      AP(84)  = GAM2IF/ RAD                                             CRASH
      AP(85)  = UMUIIP/ RAD                                             CRASH
      AP(69) = NUTA /RAD                                                CRASH
      AP(70) = TAU /60.                                                 CRASH
      RHOO   = RHOO/RAD                                                 PO14
      UMUO   = UMUO/RAD                                                 PO14
      UMU    = UMU /RAD                                                 PO14
      RHO    = RHO /RAD                                                 PO14
      A      = A * FTNM                                                 PO14
      P      = P * FTNM                                                 PO14
      RIA    = RIA * FTNM                                               PO14
      RIP    = RIP * FTNM                                               PO14
      GAM1IP = GAM1IP/ RAD                                              PO14
      GAM2IP = GAM2IP/ RAD                                              PO14
      GO TO 999                                                         CRASH
 1000 CONTINUE                                                          CRASH
      CALL PAGES(1,I,K)                                                 CRASH
      WRITE(6,2)                                                        CRASH
    2 FORMAT(1H ,9HNO IMPACT)                                           CRASH
      GO TO 1030                                                        APRL6
 1001 CONTINUE                                                          CRASH
      CALL PAGES(1,I,K)                                                 CRASH
      WRITE(6,3)                                                        CRASH
    3 FORMAT(36H APOGEE ALTITUDE BELOW EARTH SURFACE)                   APRL6
      GO TO 1030                                                        APRL6
  999 RETURN                                                            CRASH
      END                                                               CRASH
```

20 OCT 72 G.01-46

SUBRØUTINE
DCTØE

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
SQRT		F	Square root function	/SQRT	/($)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTDE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAY02	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT

```
      SUBROUTINE OCTOE(DIRCOS,EULER,KROT)                            OCTOE
C                                                                    OCTOE
C          TRANSFORMS DIRECTION COSINES INTO EULER ANGLES            OCTOE
C     REF.  GVPAT ROUTINE  OCTOE                                     OCTOE
C          KROT.ST.0  =  YAW, PITCH ROLL                             OCTOE
C          KROT.GT.  0   =  PITCH,YAW, ROLL                          OCTOE
C                                                                    OCTOE
      DIMENSION DIRCOS(3,3) ,EULER(3)                                OCTOE
      IF(KROT) 111,111,112                                           OCTOE
C                                                                    OCTOE
C          EULER ANGLES FOR ROTATION ORDER YAW, PITCH, ROLL          OCTOE
C                                                                    OCTOE
  111 EULER(2) = ATAN2(DIRCOS(1,2),DIRCOS(1,1))                      OCTOE
      EULER(3) = ATAN2(DIRCOS(2,3),DIRCOS(3,3))                      OCTOE
      EULER(1) = ATAN2(-DIRCOS(1,3),SQRT(1.- DIRCOS(1,3)*DIRCOS(1,3)))  OCTOE
      GO TO 999                                                      OCTOE
C                                                                    OCTOE
C          EULER ANGLES FOR ROTATION ORDER  PITCH, YAW, ROLL         OCTOE
C                                                                    OCTOE
  112 EULER(2) = ATAN2(-DIRCOS(1,2),SQRT(1.-DIRCOS(1,2)*DIRCOS(1,2)))  OCTOE
      EULER(3) = ATAN2(-DIRCOS(3,2),DIRCOS(2,2))                     OCTOE
      EULER(1) = ATAN2(-DIRCOS(1,3),DIRCOS(1,1))                     OCTOE
  999 RETURN                                                         OCTOE
      END                                                            OCTOE
```

111 112 999

20 OCT 72 6.01-46

SUBRØUTINE
INBVAD

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ALFMAX	α_{MAX}	M	Maximum angle of attack (DEG)	/ARCDAT/(	16)	INBVAD MODELA MODELB	M I I	ALFMAX ALFMAX ALFMAX
AMAX	x_N	I	The largest value of the first independent variable of a bivariate table.	/BICUBE/(	2)	BLICO BLYNE INBVAD	O I I	MMAX AMAX AMAX
AMIN	x_0	I	The smallest value of the first independent variable of a bivariate table.	/BICUBE/(	1)	BLICO BLYNE INBVAD	M I I	MMIN AMIN AMIN
C		O	A 32 word array containing the spline coefficients for the two bivariate functions at rectangle IRECT.	/BICUBE/(	12)	BLYNE BLYNE INBVAD	O I O	C CLOO C
IF		M	Last file in the grid in which interpolation occurred.	/BICUBE/(	3)	BLYNE INBVAD	M M	IF IF
IFMAX	N	I	Total number of files in grid.	/BICUBE/(	4)	BLICO BLYNE INBVAD	M I I	IRMAX IFMAX IFMAX
IR		M	Last rank in the grid in which interpolation occurred.	/BICUBE/(	7)	BLICO BLYNE INBVAD	M M M	IF IR IR
IREC		M	Logical record on IUNIT that contains spline coefficients for rectangle IRECT.	/BICUBE/(	11)	BLICO BLYNE INBVAD	M M M	IREC IREC IREC
IRECT		M	Grid rectangle associated with IR and IF.	/BICUBE/(	10)	BLICO BLYNE INBVAD	M M M	IRECT IRECT IRECT
IRMAX		I	Total number of ranks in grid.	/BICUBE/(	8)	BLICO BLYNE INBVAD	M I I	IFMAX IRMAX IRMAX
IUNIT		I	Logical unit number on which bicubic spline coefficients are stored for this table.	/BICUBE/(	9)	BLICO BLYNE INBVAD	M I I	IUNIT IUNIT IUNIT
T		I	A 160 word array containing logical record IREC.	/BICUBE/(	44)	BLICO BLYNE INBVAD	I I I	T T T

```
      SUBROUTINE INBVAD(ISET)                                       INBVAD
C                                                                   COMN
C        INITIALIZES  BIVARIATE  INTERPOLATION                      COMN
C                                                                   COMN
      COMMON /BICUBE/ AMIN, AMAX, IF, IFMAX, MMIN, MMAX, IR, IRMAX,  INBVAD
     1IUNIT, IRECT, IREC, C(32), T(160), F(70)                      INBVAD
      COMMON /ARCDAT/ BETA(40)                                      INBVAD
      EQUIVALENCE (ALFMAX, BETA(16))                                INBVAD
      CALL READMS(ISET + 27, AMIN, 273, 1)                          INBVAD
      IF(ALFMAX .LE. 0.) ALFMAX = 3.1415926536                      INBVAD
      IF(AMIN .GT. -ALFMAX .OR. ALFMAX .GT. AMAX) STOP 11111        INBVAD
    1 CONTINUE                                                      FIXED
      IF = (IFMAX + 1)/2                                            INBVAD
      IR = (IRMAX + 1)/2                                            INBVAD
      IRECT = IR + IRMAX*(IF - 1)                                   INBVAD
      IREC = (IRECT - 1)/5 + 2                                      INBVAD
      CALL READMS(IUNIT, T, 160, IREC)                              INBVAD
      IB = 32*(IRECT - 5*IREC + 9)                                  INBVAD
      DO 10 I = 1, 32                                               INBVAD
      J = I + IB                                                    INBVAD
   10 C(I) = T(J)                                                   INBVAD
      RETURN                                                        INBVAD
C             AIRBREATHER  ENTRY                                    FIXED
      ENTRY INBVPD                                                  FIXED
      CALL  READMS(ISET +27,AMIN,273,1)                             FIXED
      GO  TO  1                                                     FIXED
      END                                                           INBVAD
```

20 OCT 72 6.01-46

SUBRØUTINE
ISPRAT

Subroutine ISPRAT

Entry ISPIN

Purpose

Subroutine ISPRAT computes the effective ISP for the dual engine simulation associated with SSP sizing problems.

Description

The equations in ISPRAT are described in Section 7 of Vol I. Entry point ISPIN is called to initialize the vacuum thrust values and ISP ratio of the two engines.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
SQ		I	A synthesis data array (37,5) that contains the flyback data and some injection quantities	/SIZING/(	74)	ENVPRA	A	SQ
						FLYBKP	A	SQ
						ISPRAT	I	SQ
						PDOC	I	SQ
						PRITVA	I	SQ
						RANGE	A	SQ
						REU3	O	SQ
						SIZE.	O	SQ
						SIZEMR	A	SQ
						SIZIN	A	SQ
						STAU	I	SQ
						SUMOUT	A	SQ
						TAMPAR	O	SQ
						TAMPER	A	SQ
						THRUST	A	SQ
						TRTOSZ	A	SQ
						VEHOF	A	SQ
						WTVOL	A	SQ

```
      SUBROUTINE ISPRAT( TRAT,ISR,ISRT,ISRTT,I)                         ISPRAT
C          COMPUTE  MEAN  ISP RATIO  FOR  DUAL  ENGINE SIMULATION        ISPRAT
C                         DURING THROTTLING                             ISPRAT
C          WHEN ENGINES  HAVE  DIFFERENT  ISP VALUES                    ISPRAT
C                                                                       ISPRAT
C                HAS    TWO  ENTRY  POINTS                              ISPRAT
C                ISPIN,  INITIALIZES  DATA  AT ARC  CORNER POINT        ISPRAT
      REAL  ISR,ISRT,ISRTT,KPRIME                                       ISPRAT
      REAL MUB, MUO, ISPB, ISPO, IDVEL,WWB,WD                           SIZING
      COMMON /SIZING/                                                   SIZING
C           PHASE II SIZING PARAMERERS                                  SIZING
     *TZ,      VV(3),     QP(14),   EROR,    PZ(5),     VQ,       SW(20), SIZING
     *SV(28),  SQ(37,5),  SE(11),   TLAT,    TLNG,                      SIZING
C          PHASE I SIZING PARAMERERS                                    SIZING
     *WBO,     WLOD,      DWEB,     DWEO,     TOLWT,    WPB,      TWRAT2, SIZING
     *BK1,     BK2,       BK3,      BK4,      ISIZE,    TRAFLG,   TWRATO, SIZING
     *OK1,     OK2,       OK3,      OK4,      PRFLG,    IPASS,    IPSMAX, SIZING
     *AEXIT,   TVACO,     WD,       WFO,      IDVEL,    ISPO,     ISPB,   SIZING
     *XPL,     TVACB,   WWB,        WEO,      WEB,      WD,       WLO,    SIZING
     *DVO,      DVB,       MUB,      MUO,      VSTG,     WPO              SIZING
     *,JTYP,    BECO ,    BSTG ,    ORBI,    ITNOW , ITNOW  ,            SIZING
     * SVOPSQ  , SVDCON  ,IHUNT   ,IOPSTG  ,ISZD(19)                    UH
      EQUIVALENCE (RV1,TV1),(RV2,TV2)                                   JULY28
      TRAD= TRAT*.01                                                    ISPRAT
      DEN = (KPRIME  + TRAD -1.)*RV1 + TRAD*RV2                         ISPRAT
      ISR =  TRAD *(KPRIME*TV1 + TV2)/DEN                               ISPRAT
      IF(I.EQ.1) GO TO 103                                              ISPRAT  103
      ISRT = ISR*( 1./TRAD -(RV1+TV2)/DEN )                             ISPRAT
      IF(I.EQ.2) GO TO 102                                              ISPRAT  102
      ISRTT = ISRT**2/ISR +ISR*(-1./TRAD**2+((TV1+TV2)/DEN)**2)         JULY28
      ISRTT = 100.*ISRTT                                                ISPRAT
  102 ISRT  = 100.*ISRT                                                 ISPRAT
  103 ISR   = 100.*ISR                                                  ISPRAT
      RETURN                                                            ISPRAT
C           INITIALIZATION ENTRY                                        ISPRAT
      ENTRY ISPIN                                                       ISPRAT
C             TV1 IS THE UNTHROTTLE ENGINE ,TV2 IS THROTTLED  THRUST(VAC) ISPRAT
C             KPRIME  IS  THE  RATIO  ISP-2 / ISP-1   (VACUUM ONLY)     ISPRAT
      TV1=SQ(31,1)                                                      ISPRAT
      TV2=SQ(31,2)                                                      ISPRAT
      KPRIME=SQ(31,3)                                                   ISPRAT
      RETURN                                                            ISPRAT
      END                                                               ISPRAT
```

MATADD

```
      SUBROUTINE MATADD(S, A, B, N, M)                                  MATADD
C                                                                       MATADD
C     THIS SUBPROGRAM PERFORMS THE DOUBLE PRECISION MATRIX ADDITION     MATADD
C     S=A+B, WHERE N IS THE NUMBER OF ROWS IN THE A (B) MATRIX AND M IS MATADD
C     THE NUMBER OF COLUMNS IN THE A (B) MATRIX.                        MATADD
C                                                                       MATADD
      DIMENSION A(1), B(1), S(1)                                        MATADD
C                                                                       MATADD
      J = N*M                                                           MATADD
      DO 1 I = 1, J                                                     MATADD
    1 S(I) = A(I) + B(I)                                                MATADD
      RETURN                                                            MATADD
      END                                                               MATADD
```

20 OCT 72 6.01-46

MATMLT

```
      SUBROUTINE MATMLT(C, A, B, MM, N, L)                       MATMLT
      DIMENSION A(1),B(1),C(1)                                   MATMLT
      L11 = MM*N                                                 MATMLT
      M11 = MM*L                                                 MATMLT
      DO 20 I = 1,MM                                             MATMLT
      L2 = 1                                                     MATMLT
      DO 20 J = I,M11,MM                                         MATMLT
      SUM  = 0.0                                                 MATMLT
      DO 10 K = I,L11,MM                                         MATMLT
      SUM  = SUM  + A(K)*B(L2)                                   MATMLT
   10 L2 = L2 + 1                                                MATMLT
   20 C(J) = SUM                                                 MATMLT
      RETURN                                                     MATMLT
      END                                                        MATMLT
```

20 OCT 72 6.01-46

PAGES

Subroutine PAGES

Purpose

PAGES keeps track of page and line count and returns a flag to indicate when a page should be thrown. PAGES is called from print.

```
      SUBROUTINE PAGES(NOLINE,IPAGE,ISKIP)                              PAGES
C                                                                       PAGES
C          SUBROUTINE TO ASSESS IF A PRINT PO-NT WILL                   PAGES
C         FIT ON PAGE OR IF A NEW PAGE SHOULD BE STARTED                PAGES
C                                                                       PAGES
      IF(NOLINE.EQ.0) GO TO 5                                           NOS
      IF(LINENO+NOLINE.LE.66) GO TO 10                                  NOS
C                                                                       PAGES
C         START NEW PAGE                                                PAGES
C                                                                       PAGES
    5 CONTINUE                                                          NOS
      LINENO = 0                                                        SEP18
      ISKIP = 1                                                         PAGES
      IPAGE  = IPAGE + 1                                                PAGES
      GO TO 999                                                         PAGES
C                                                                       PAGES
C          USE CURRENT PAGE                                             PAGES
C                                                                       PAGES
   10 LINENO = LINENO + NOLINE                                          APRL6
      ISKIP = 0                                                         PAGES
      IPAGE = IPAGE                                                     PAGES
  999 RETURN                                                            PAGES
      END                                                               PAGES
```

20 OCT 72 6.01-46

74

SUBRØUTINE PAYLØD

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DWEB		I	Sensitivity of booster stage weight to propellant weight (lb/lb)	/SIZING/(	274)	PAYLOD	I	DWEB
						SIZOUT	I	DWEB
						SIZ1	M	DWEB
						SIZ2	M	DWEB
						STAU	I	DWEB
						WTORP	M	DWEB
DWEO		I	Sensitivity of orbiter stage weight to propellant weight (lb/lb)	/SIZING/(	275)	PAYLOD	I	DWEO
						SIZOUT	I	DWEO
						SIZ1	M	DWEO
						SIZ2	M	DWEO
						SIZ4	M	DWEO
						WTORP	M	DWEO
WBO		O	Booster burnout weight (lb)	/SIZING/(	272)	GEINP	M	SIZ
						PAYLOD	O	WBO
						SIZE	I	DAT
						SIZOUT	I	WBO
						SIZ1	M	WBO
						SIZ2	M	WBO
						SIZ4	M	WBO
						TAMPAR	I	WBO
WEB		I	Booster stage weight (lb)	/SIZING/(	304)	PAYLOD	I	WEB
						SIZOUT	I	WEB
						SIZ1	M	WEB
						SIZ2	M	WEB
						SIZ4	I	WEB
						TAMPAR	I	WEB
						WTORP	M	WEB
WEO		I	Orbiter stage weight (lb)	/SIZING/(	303)	PAYLOD	I	WEO
						SIZOUT	I	WEO
						SIZ1	M	WEO
						SIZ2	M	WEO
						SIZ3	I	WEO
						SIZ4	M	WEO
						TAMPAR	I	WEO
						WTORP	M	WEO
WFO		M	Orbiter burnout weight (lb)	/SIZING/(	296)	PAYLOD	M	WFO
						SIZOUT	I	WFO
						SIZ1	O	WFO
						SIZ2	O	WFO
						SIZ3	O	WFO
						SIZ4	M	WFO
						TAMPAR	I	WFO
						TRTOSZ	O	WFO
WLO		O	Booster liftoff weight (lb)	/SIZING/(	306)	PAYLOD	O	WLO
						SIZE	I	WLO
						SIZOUT	I	WLO
						SIZ1	I	WLO
						SIZ2	M	WLO
						SIZ4	M	WLO
						SIZ5	O	WLO
						TAMPAR	I	WLO
WO		O	Initial orbiter weight (lb)	/SIZING/(	305)	PAYLOD	O	WO
						SIZOUT	I	WO
						SIZ1	M	WO
						SIZ2	M	WO
						SIZ3	M	WO
						SIZ4	M	WO
						TAMPAR	I	WO

76

PAYLOD

```
      SUBROUTINE PAYLOD(PL,DP1,DP3,W1,W2)                                PAYLOD
C                                                                        PAYLOD
C         OPTIMAL STAGING OPTION PAYLOAD CALCULATIONS                    PAYLOD
C                                                                        PAYLOD
      REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                            SIZING
      COMMON /SIZING/                                                    SIZING
C          PHASE II SIZING PARAMERERS                                    SIZING
     *TZ,      VV(3),     DP(14),   EROR,     PZ(5),    VQ,      SW(20), SIZING
     *SV(28),  SQ(37,5),  SE(11),   TLAT,     TLNG,                      SIZING
C         PHASE I SIZING PARAMERERS                                      SIZING
     *WBO,     WLOO,      DWEB,     DWEO,     TOLWT,    WPB,     TWRAT2, SIZING
     *BK1,     BK2,       BK3,      BK4,      ISIZE,    TRAFLG,  TWRATO, SIZING
     *OK1,     OK2,       OK3,      OK4,      PRFLG,    IPASS,   IPSMAX, SIZING
     *AEXIT,   TVACO,     NO,       WFO,      IDVEL,    ISPO,    ISPB,   SIZING
     *XPL,     TVACB,    NNB,       WEO,      WEB,      WO,      WLO,    SIZING
     *DVO,      DVB,       MUB,      MUO,      VSTG,     WPO             SIZING
     *,JTYP,    BECO ,    BSTG ,    ORBI,    ITNBW , ITNOW  ,            SIZING
     * SVDPSQ  , SVDCON  ,IHUNT   ,IOPSTG  ,ISZD(19)                     UH
      WO = W1                                                            PAYLOD
      WFO = W2                                                           PAYLOD
      WBO = W1 + WEB                                                     PAYLOD
      WLO = VV(1)                                                        PAYLOD
C                                                                        PAYLOD
C         PAYLOAD WEIGHT                                                 PAYLOD
C                                                                        PAYLOD
      PL=WFO-WEO                                                         PAYLOD
C                                                                        PAYLOD
C         PAYLOAD SENSITIVITIES TO                                       PAYLOD
C                                                                        PAYLOD
C         1.  ORBITER FINAL WEIGHT                                       PAYLOD
C                                                                        PAYLOD
      DP1=1.+DWEO                                                        PAYLOD
C                                                                        PAYLOD
C         2.  ORBITER INITIAL WEIGHT                                     PAYLOD
C                                                                        PAYLOD
C                                                                        PAYLOD
C         3   BOOSTER FINAL WEIGHT                                       PAYLOD
C                                                                        PAYLOD
      DP3=-DWEO*(1.+DWEB)                                                PAYLOD
C                                                                        PAYLOD
      RETURN                                                             PAYLOD
      END                                                                PAYLOD
```

20 OCT 72 G.01-46

SUBRØUTINE
PRINT

78

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ID		I	A four word array containing the basic deck, reference run, case and part case numbers in that order.	/GLOBAL/(	21)	BLICO	I	ID
						FRENCH	I	ID
						GEINP	I	ID
						PADS1	O	ID
						PRINT	I	ID
						SDINP	I	ID
						TOPM	I	ID
						VEHDF	I	ID
IPFLG1		I	IPFLG1≠0 supresses print-out of velocity losses and inertial Euler angles.	/GLOBAL/(	69)	FNTG	I	IPFLG1
						OUT	I	IPFLG1
						PDBC	I	IPFLG1
						PRINT	I	IPFLG1
						TRTOSZ	O	IPFLG1
IPFLG2		I	IPFLG2≠0 supresses print-out of orbital parameters.	/GLOBAL/(	70)	PRINT	I	IPFLG2
						TRTOSZ	O	IPFLG2
IPFLG3		I	IPFLG3≠0 supresses print-out of impact data.	/GLOBAL/(	71)	OUT	I	IPFLG3
						PRINT	I	IPFLG3
						TRTOSZ	O	IPFLG3
IPFLG4		I	IPFLG4≠0 supresses print-out of inertial Cartesian coordinates.	/GLOBAL/(	72)	PRINT	I	IPFLG4
.UN06.		O	File of all output data	/.UN06./($	)	BLICO	O	.UN06.
						BNDRYC	O	.UN06.
						CRASH	O	.UN06.
						FRENCH	O	.UN06.
						FXDAT	O	.UN06.
						GEINP	O	.UN06.
						HUNT	O	.UN06.
						INEDIT	O	.UN06.
						ITER8	O	.UN06.
						MODELA	O	.UN06.
						MOMJ	O	.UN06.
						MPSI	O	.UN06.
						OUT	O	.UN06.
						PAYO2	O	.UN06.
						PRINT	O	.UN06.
						PRINTV	O	.UN06.
						PRINTW	O	.UN06.
						PRITEQ	O	.UN06.
						PRITVA	O	.UN06.
						PROPIN	O	.UN06.
						PROTHR	O	.UN06.
						PRWTSM	O	.UN06.
						RANGE	O	.UN06.
						S	O	.UN06.
						SDINP	O	.UN06.
						SIZE	O	.UN06.
						SIZIN	O	.UN06.
						SIZOUT	O	.UN06.
						SOLVE	O	.UN06.
						SPLICO	O	.UN06.
						SPLIZ	O	.UN06.
						SPLYNE	O	.UN06.
						SSSP	O	.UN06.
						STAU	O	.UN06.
						STPIT	O	.UN06.
						SUMOUT	O	.UN06.
						TABIN	O	UN06.
						TEST	O	.UN06.
						VEHDF	O	.UN06.
						WTSCH	O	.UN06.
						WTVOL	O	.UN06

```
      SUBROUTINE PRINT(ITER,N,M,K)                                     PRINT
      COMMON/GLOBAL/                                                   GLOBAL
     *GR         ,ER        ,OMGZ      ,XLAMRF    ,YMURF    ,LUM       GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC      ,NBRAN     ,NFARC    ,ID(4)     GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                             GLOBAL
     *,GM          ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)  GLOBAL
     *,ITPSO  ,KSOL     ,KGLOBL(8)                                     RETAP
      DIMENSION          LABL1(3),LABL2(3),LABL3(3),LABL4(3),LABEL(3)  PRINT
      COMMON /PRINT/  A(100)                                           PRINT
      DATA LABL1/6H  SD I,6HTERATI,6HON NO /,                          PRINT
     2     LABL2/6H      ,6HSD SOL,6HUTION /,                          PRINT
     3     LABL3/6H  QL I,6HTERATI,6HON NO /,                          PRINT
     4     LABL4/6H      ,6HQL SOL,6HUTION /                           PRINT
C     JLAG=-1  HEADER JUST PRINTED                                     PRINT
C     JLAG=+1  HEADER TO BE PRINTED                                    PRINT
C     N =1     SOLUTION TRAJ                                           PRINT
C     N =0     TRIAL TRAJ                                              PRINT
C     M = 1    QL TRAJ                                                 PRINT
C     M = 0    SD TRAJ                                                 PRINT
C         K - NON ZERO MEANS USE FIRST PASS LOGIC                      PRINT
C       IPFLG1 =1   NO PRINT  ROW 9                                    PRINT
C       IPFLG2 =1   NO PRINT  ROW 10 AND 11                            PRINT
C       IPFLG3 =1   NO PRINT  ROW 12 AND 13                            PRINT
C       IPFLG4 =1   NO PRINT  ROW 14                                   PRINT
C       L= NUMBER OF LINES                                             APRL6
C                                                                      PRINT
      IF(K NE.0)  GO TO 60                                             PRINT
      IF(N.EQ.0)     GO TO 150                                         PRINT
C                                                                      PRINT
C          COUNT LINES                                                 PRINT
  151 CONTINUE                                                         NOS
      L=10                                                             APRL6
      IF(IPFLG1.EQ.0) L=L+1                                            PRINT
      IF(IPFLG2.EQ.0) L=L+2                                            PRINT
      IF(IPFLG3.EQ.0) L=L+2                                            PRINT
      IF(IPFLG4.EQ.0) L=L+1                                            PRINT
      CALL PAGES(L,IPAGE,ISKIP)                                        PRINT
      IF(ISKIP.EQ.1)  GO TO 400                                        PRINT
C          PRINT DATA                                                  PRINT
  405 KJ=0                                                             NOS
      DO 7 J=1,36,7                                                    NOS
      KJ=KJ+1                                                          NOS
      KK=J+6                                                           APR72
      WRITE(6,12) KJ,(A(I),I=J,KK)                                     NOS
    7 CONTINUE                                                         APRL6
      L7=7                                                             APRL6
      WRITE(6,12) L7,(A(I),I=43,47),A(92),A(93)                        APRL6
      L8=8                                                             APRL6
      WRITE(6,12) L8,(A(I),I=48,54)                                    APRL6
      L9= 9                                                            APRL6
      L10=10                                                           APRL6
      L11=11                                                           APRL6
      L12=12                                                           APRL6
      L13=13                                                           APPL6
      L14= 14                                                          APRL6
      IF(IPFLG1.EQ.0) WRITE(6,12) L9 ,(A(I), I=55,61)                  APRL6
      IF(IPFLG2.EQ.0) WRITE(6,12) L10,(A(I), I=62,68)                  APRL6
      IF(IPFLG2.EQ.0) WRITE(6,12) L11 ,(A(I), I=69,75)                 APR72
      IF(IPFLG3.EQ.0) WRITE(6,15) L12,(A(I), I=76,81)                  APR72
      IF(IPFLG3.EQ 0) WRITE(6,16) L13,(A(I), I=82,85)                  APR72
      IF(IPFLG4.EQ.0) WRITE(6,15) L14,(A(I), I=86,91)                  APR72
       WRITE (6,101)                                                   PRINT
      GO TO 999                                                        PRINT
  150 CONTINUE                                                         PRINT
      L=10                                                             APRL6
      CALL PAGES(L,IPAGE,ISKIP)                                        PRINT
      IF(ISKIP EQ.1)  GO TO 400                                        PRINT
  403 CONTINUE                                                         PRINT
       KJ=0                                                            NOS
      DO 8 J=1,36,7                                                    NOS
      KJ=KJ+1                                                          NOS
      KK=J+6                                                           APR72
      WRITE(6,12) KJ,(A(I),I=J,KK)                                     NOS
    8 CONTINUE                                                         APRL6
```

```
      L7=7                                                      APRL6
      WRITE(6,12) L7,(A(I),I=43,47),A(92),A(93)                 APRL6
      L8=8                                                      APRL6
      WRITE(6,12) L8,(A(I),I=48,54)                             APRL6
       WRITE (6,101)                                            PRINT
      GO TO 999                                                 PRINT
C                                                               PRINT
C          PRINT HEADER  ID                                     PRINT
C                                                               PRINT
  400 WRITE(6,100)                                              PRINT
      JLAG =-JLAG                                               PRINT
  404 IF(N.EQ 0 AND.M.EQ 0) WRITE(6,1)ID(3),LABL1,ITER,IPAGE,IDATE PRINT
      IF(N.EQ 0 AND.M EQ.1) WRITE(6,1)ID(3),LABL3,ITER,IPAGE,IDATE PRINT
      IF(N.EQ.1 AND.M.EQ.0) WRITE(6,11)ID(3),LABL2,IPAGE,IDATE  PRINT
      IF(N.EQ 1 AND.M EQ.1) WRITE(6,11)ID(3),LABL4,IPAGE,IDATE  PRINT
      L=2                                                       NOS
      CALL PAGES(L,IPAGE,ISKIP)                                 APRL6
  401 IF(JLAG.GE.0)  GO TO 50                                   PRINT
      IF(N EQ.0) GO TO 150                                      NOS
      GO TO 151                                                 NOS
C                                                               PRINT
C          OUTPUT HEADER FORMAT PRINT REQUIRED                  PRINT
C                                                               PRINT
   50 IF(N EQ.0) GO TO 51                                       PRINT
C                                                               PRINT
C          CHECK FOR SURPRESSED ROWS                            PRINT
C                                                               PRINT
      WRITE(6,2)                                                PRINT
      IF(IPFLG1 EQ 0) WRITE(6,3)                                PRINT
      IF(IPFLG2 EQ.0) WRITE(6,4)                                PRINT
      IF(IPFLG3 EQ.0) WRITE(6,5)                                PRINT
      IF(IPFLG4 EQ.0) WRITE(6,6)                                PRINT
      WRITE(6,101)                                              PRINT
      L=15                                                      NOS
      IF(IPFLG1 NE.0)  L = L -1                                 PRINT
      IF(IPFLG2.NE.0)  L = L -2                                 PRINT
      IF(IPFLG3.NE.0)  L = L -2                                 PRINT
      IF(IPFLG4.NE.0)  L = L -1                                 PRINT
      CALL PAGES(L,IPAGE,ISKIP)                                 NOS
      GO TO 151                                                 NOS
   51 WRITE(6,2)                                                PRINT
       WRITE (6,101)                                            PRINT
      L=10                                                      APRL6
      CALL PAGES(L,IPAGE,ISKIP)                                 APRL6
      GO TO 150                                                 NOS
      ENTRY PRORPA                                              PRINT
C                                                               PRINT
C          PRINT ORBIT PARAMETERS                               PRINT
C                                                               PRINT
      L=5                                                       APRL6
      CALL PAGES(L,IPAGE,ISKIP)                                 APRL6
      WRITE(6,4)                                                PRINT
      WRITE(6,101)                                              PRINT
      L10=10                                                    APRL6
      L11=11                                                    APRL6
      WRITE(6,12) L10,(A(I),I=62,68)                            APRL6
      WRITE(6,12) L11,(A(I),I=69,75)                            APRL6
      GO TO 999                                                 PRINT
C                                                               PRINT
C          FIRST PASS LOGIC                                     APRL6
C                                                               PRINT
   60 CONTINUE                                                  PRINT
      CALL DATE1(MON,IDAY,IYEAR)                                P014
      IDATE=IYEAR+IDAY*100+MON*10000                            P014
      JLAG = -1                                                 PRINT
      IPAGE = 0                                                 PRINT
      L=0                                                       APRL6
C                                                               APRL6
C     INITIATE LINE COUNT                                       APRL6
C                                                               APRL6
      CALL PAGES(L,IPAGE,ISKIP)                                 APRL6
      GO TO 400                                                 PRINT
C                                                               PRINT
```

```
C          HEADER FORMAT                                                  PRINT
C                                                                         PRINT
    1 FORMAT(1H ,5HCASE ,I3,7X,3A6,I3,20X,6H PAGE ,I3,50X,5HDATE ,I9 )    PRINT
   11 FORMAT(1H ,5HCASE,I3,7X,3A6,23X,6H PAGE ,I3,50X,5HDATE ,I9)         PRINT
C                                                                         PRINT
C          SHORT HEADER FORMAT ROWS 1 THRU 8                              PRINT
C                                                                         PRINT
    2 FORMAT(3H  1    ,14X,4HTIME,10X,8HALTITUDE,6X,12HREL VELOCITY,4X,   PRINT
     114HREL PATH ANGLE,7X,11HREL AZIMUTH,5X,13HREL LONGITUDE,10X,        PRINT
     28HLATITUDE/3H  2    ,10X,8HARC TIME,12X,6HWEIGHT,6X,12HINR VELOCITY PRINT
     3,4X,14HINR PATH ANGLE,7X,11HINR AZIMUTH,5X,13HINR LONGITUDE,5X,     PRINT
     413HCROSS RNG(NM)/3H  3    ,8X,10HPHASE TIME,14X,4HMASS,             PRINT
     54X,14HIDEAL VELOCITY,                                               PRINT
     59X,9HHEAT LOAD,9X,9HHEAT RATE,9X,9HRANGE(NM),6X,12HDOWN RNG(NM)/    PRINT
     63H  4    ,9X,9HRE NUMBER,6X,12HAMB PRESSURE,5X,13HATMOS DENSITY,    PRINT
     77X,11HSPEED SOUND,7X,    11HMACH NUMBER,8X,10HLIFT COEFF,8X,        APRL6
     810HDRAG COEFF/3H  5    ,13X,5HALPHA,8X,10HBANK ANGLE,6X,            PRINT
     912HBLEND FACTOR,5X,13HDYNAMIC PRESS,7X,11HAERO MOMENT,14X,4HLIFT,   PRINT
     *14X,4HDRAG/3H  6    ,12X,6HTHRUST,9X,9HCOSTATE V,5X,13HCOSTATE GAMM PRINT
     1A,7X,11HCOSTATE AZI,7X,11HCOSTATE ALT,7X,11HCOSTATE LAT,6X,         PRINT
     212HCOSTATE LONG/3H  7    ,6X,12HSPEC IMPULSE,6X,12HCOSTATE MASS,    PRINT
     33X,15HCOSTATE HEATING,5X,13HCOSTATE YYYYY,7X,11HHAMILTONIAN         APRL6
     *,5X,13HSTEERING ELEV,6X,12HSTEERING AZI/3H  8                       PRINT
     4    ,6X,12HGIMBAL ANGLE,9X,9HAXIAL ACC,8X,10HNORMAL ACC,9X,         PRINT
     59HTOTAL ACC,9X,9HREL PITCH,11X,7HREL YAW,10X,8HREL ROLL)            PRINT
C                                                                         PRINT
C          ROW 9 FORMAT                                                   PRINT
C                                                                         PRINT
    3 FORMAT(1H ,2H 9,9X,9HDRAG LOSS,6X,12HGRAVITY LOSS,8X,               PRINT
     110HALPHA LOSS,3X,15HBACK PRESS LOSS,9X,9HINR PITCH,11X,             PRINT
     27HINR YAW,10X,8HINR ROLL)                                           PRINT
C                                                                         PRINT
C          ROW 10 AND 11 FORMAT (ORBIT ELEMENTS)                          PRINT
C                                                                         PRINT
    4 FORMAT(1H ,2H10,5X,13HSEMI AXIS(NM),6X,12HECCENTRICITY,7X,          PRINT
     111HINCLINATION,4X,14HASCENDING NODE,7X,11HARG PERIGEE,4X,           PRINT
     214HAPOGEE RAD(NM),3X,15HPERIGEE RAD(NM)/3H 11    ,6X,               PRINT
     312HTRUE ANOMALY,4X,14H   PERIOD(MIN),12X,6HENERGY,10X,              PRINT
     48HMOMENTUM,2X,16HSEMI LAT REC(NM),3X,15HAPOGEE VELOCITY,2X,         PRINT
     516HPERIGEE VELOCITY)                                                PRINT
C                                                                         PRINT
C          ROW 12 AND 13 FORMAT (IIP)                                     PRINT
C                                                                         PRINT
    5 FORMAT(3H 12    ,5X,13HIMP TIME(MIN),5X,13HIMP RANGE(NM),4X,        PRINT
     114HAZIMUTH TO IMP,24X,12HIMP VELOCITY,4X,14HIMP PATH ANGLE,6X,      PRINT
     212HIMP LATITUDE/3H 13    ,2X,16HTIME TO IMP(MIN),2X,                PRINT
     316HRANGE TO IMP(NM),55X,17HIMP AZIMUTH ANGLE,5X,13HIMP LONGITUDE)   PRINT
C                                                                         PRINT
C          ROW 14 CARTESIAN INERTIAL COORDINATES                          PRINT
C                                                                         PRINT
    6 FORMAT(3H 14    ,8X,10HCART-X INR,8X,10HCART-Y INR,8X,              PRINT
     110HCART-Z INR,23X,13HCART-XDOT INR,5X,13HCART-YDOT INR,5X,          PRINT
     213HCART-ZDOT INR)                                                   PRINT
   12 FORMAT(1H ,I2,7E18.7)                                               APRL6
   15 FORMAT(1H ,I2,3E18.7,18X,3E18.7)                                    APRL6
   16 FORMAT(1H ,I2,2E18.7,54X,2E18.7)                                    APRL6
  100 FORMAT(1H1)                                                         PRINT
  101 FORMAT(1H0)                                                         PRINT
   14 FORMAT(1H0,7E18.7)                                                  PRINT
  999 RETURN                                                              PRINT
      END                                                                 PRINT
```

20 OCT 72 G.01-46

PRODUT

```
      SUBROUTINE PRODUT(A,B,C)                                          PRODUT
C                                                                       PRODUT
C         3 BY 3  MATRIX PRODUCT   A*B = C                              PRODUT
C                                                                       PRODUT
      DIMENSION A(9), B(9), C(9)                                        PRODUT
      C(1) =  A(1)*B(1) + A(4)*B(2) + A(7)*B(3)                         PRODUT
      C(2) =   A(2)*B(1) + A(5)*B(2) + A(8)*B(3)                        PRODUT
      C(3) =  A(3)*B(1) + A(6)*B(2) + A(9)*B(3)                         PRODUT
      C(4) =  A(1)*B(4) + A(4)*B(5) + A(7)*B(6)                         PRODUT
      C(5) =  A(2)*B(4) + A(5)*B(5) + A(8)*B(6)                         PRODUT
      C(6) =  A(3)*B(4) + A(6)*B(5) + A(9)*B(6)                         PRODUT
      C(7) =  A(1)*B(7) + A(4)*B(8) + A(7)*B(9)                         PRODUT
      C(8) =  A(2)*B(7) + A(5)*B(8) + A(8)*B(9)                         PRODUT
      C(9) =  A(3)*B(7) + A(6)*B(8) + A(9)*B(9)                         PRODUT
      RETURN                                                            PRODUT
      END                                                               PRODUT
```

20 OCT 72 6.01-96

SUBRØUTINE
SPLIZ

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
LOCF		I	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the last value of the independent variable of the corresponding table	/TABLE /(	1401)	SPLICO	M	LOCF
						SPLICO	M	Z
						SPLIZ	I	LOCF
						SPLIZ	I	Z
						SPLYNE	I	LOCF
						SPLYNE	I	Z
						THRUP	O	Z
LOCI		I	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that points to the initial value of the independent variable of the corresponding table A zero entry indicates table not input	/TABLE /(	1)	SPLICO	M	LOCI
						SPLICO	M	X
						SPLIZ	I	LOCI
						SPLIZ	I	X
						SPLYNE	I	LOCI
						SPLYNE	I	X
						THRUP	I	LOCI
						THRUP	O	X
LOCL		M	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that indicates the last interval in which interpolation of the corresponding table occured.	/TABLE /(	701)	SPLICO	O	LOCL
						SPLICO	M	Y
						SPLIZ	M	LOCL
						SPLIZ	I	Y
						SPLYNE	M	LOCL
						SPLYNE	I	Y
						THRUP	O	Y
NT		I	Largest univariant table number in this case.	/GLOBAL/(	66)	SPLICO	M	NT
						SPLIZ	I	NT
						SPLYNE	I	NT
X		I	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the initial value of the independent variable of the corresponding table. A zero entry indicates table not input.	/TABLE /(	1)	SPLICO	M	LOCI
						SPLICO	M	X
						SPLIZ	I	LOCI
						SPLIZ	I	X
						SPLYNE	I	LOCI
						SPLYNE	I	X
						THRUP	I	LOCI
						THRUP	O	X
Y		I	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that indicates the last interval in which interpolation of the corresponding table occured.	/TABLE /(	701)	SPLICO	O	LOCL
						SPLICO	M	Y
						SPLIZ	M	LOCL
						SPLIZ	I	Y
						SPLYNE	M	LOCL
						SPLYNE	I	Y
						THRUP	O	Y
Z		I	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the last value of the independent variable of the corresponding table.	/TABLE /(	1401)	SPLICO	M	LOCF
						SPLICO	M	Z
						SPLIZ	I	LOCF
						SPLIZ	I	Z
						SPLYNE	I	LOCF
						SPLYNE	I	Z
						THRUP	O	Z

58


```
      SUBROUTINE SPLIZ(IT,T,F,DFDX)                                   SPLIZ
      COMMON /TABLE/ TABLE(2100) /GLOBAL/ G(66)                       SPLIZ
      DIMENSION X(1), Y(1), Z(1), LOCI(1), LOCL(1), LOCF(1)           SPLIZ
      EQUIVALENCE (X, TABLE), (Y, TABLE(701)), (Z, TABLE(1401)),      SPLIZ
     1(LOCI, X), (LOCL, Y), (LOCF, Z), (NT, G(66))                    SPLIZ
      DATA SIXTH/17155252525252525253B/                               SPLIZ
      DATA SPLINT /6HSPLINE/                                          SPLIZ
   10 FORMAT(1H0, 22HTHE TABLE NO. IN LOC. O6 17H IS OUT OF RANGE.)   SPLIZ
   20 FORMAT(1H0, 22HTHE TABLE NO. IN LOC. O6 20H HAS NOT BEEN INPUT.) SPLIZ
      IF(IT .LE. NT) GO TO 30                                         SPLIZ
      ITLOC = XLOCF(IT)                                               SPLIZ
      WRITE(6,10) ITLOC                                               SPLIZ
      CALL STPIT (6)                                                  SPLIZ
   30 IF(IT .LE. 0) GO TO 40                                          SPLIZ
      II = LOCI(IT)                                                   SPLIZ
      IF(II .GT. 0) GO TO 50                                          SPLIZ
      ITLOC = XLOCF(IT)                                               SPLIZ
      WRITE(6,20) ITLOC                                               SPLIZ
      CALL STPIT (6)                                                  SPLIZ
   40 F = 0.                                                          SPLIZ
      DFDX = 0.                                                       SPLIZ
      RETURN                                                          SPLIZ
   50 IF = LOCF(IT)                                                   SPLIZ
      IF(IF - II - 1) 60,70,80                                        SPLIZ
   60 F = Y(II)                                                       SPLIZ
      DFDX = 0.                                                       SPLIZ
      RETURN                                                          SPLIZ
   70 Y1 = Y(II)                                                      SPLIZ
      X1 = X(II)                                                      SPLIZ
      DFDX=(Y(IF) - Y1)/(X(IF) - X1)                                  SPLIZ
      F = Y1 + (T - X1)*DFDX                                          SPLIZ
      RETURN                                                          SPLIZ
   80 IF(T .GT. X(II)) GO TO 90                                       SPLIZ
      Y1 = Y(II)                                                      SPLIZ
      X1 = X(II)                                                      SPLIZ
      IIP1 =  II + 1                                                  SPLIZ
      DEL = X(IIP1) - X1                                              SPLIZ
      DFDX=(Y(IIP1) - Y1)/DEL - SIXTH*DEL*Z(IIP1)                     SPLIZ
      F = Y1 + (T - X1)*DFDX                                          SPLIZ
      RETURN                                                          SPLIZ
   90 IF(T .LT. X(IF)) GO TO 100                                      SPLIZ
      IFM1 = IF - 1                                                   SPLIZ
      Y1 = Y(IF)                                                      SPLIZ
      X1 = X(IF)                                                      SPLIZ
      DEL = X1 - X(IFM1)                                              SPLIZ
      DFDX=(Y1 - Y(IFM1))/DEL+ SIXTH*Z(IFM1)*DEL                      SPLIZ
      F = Y1 + (T - X1)*DFDX                                          SPLIZ
      RETURN                                                          SPLIZ
  100 IL = LOCL(IT)                                                   SPLIZ
      IF(T - X(IL)) 110,150,120                                       SPLIZ
  110 IL = IL - 1                                                     SPLIZ
      IF(T - X(IL)) 110,140,140                                       SPLIZ
  120 IS = IL + 1                                                     SPLIZ
      DO 130 I = IS, IF                                               SPLIZ
      IF(T - X(I ) .LT. 0.) GO TO 140                                 SPLIZ
  130 IL = I                                                          SPLIZ
  140 LOCL(IT) = IL                                                   SPLIZ
  150 ILP1 = IL + 1                                                   SPLIZ
      DX1 = T - X(IL)                                                 SPLIZ
      DX2 = X(ILP1) - T                                               SPLIZ
      Y1 = Y(IL)                                                      SPLIZ
      Y2 = Y(ILP1)                                                    SPLIZ
      Z1 = Z(IL)                                                      SPLIZ
      Z2 = Z(ILP1)                                                    SPLIZ
      DEL = DX1 + DX2                                                 SPLIZ
      TEMP1= Z2*DX1 + Z1*DX2                                          SPLIZ
      TEMP2 = SIXTH*DEL                                               SPLIZ
      F = ((Y2*DX1 + Y1*DX2) + SIXTH*(Z2*DX1**3 + Z1*DX2**3))/DEL     SPLIZ
     1 - TEMP2*TEMP1                                                  SPLIZ
```

29 OCT 72 G.01-46

```
      DFDX = ((Y2 - Y1) + .5*(Z2*DX1**2 - Z1*DX2**2))/DEL -TEMP2*(Z2-Z1) SPLIZ
      RETURN                                                             SPLIZ
      END                                                                SPLIZ
```

20 OCT 72 6.01-46

SUBRØUTINE TABIN

SUBROUTINE TABIN SPECIFICATIONS

I. DESCRIPTION

Purpose:

Inputs a new case of data or inputs data for a different part case of the current case.

Comments:

This subroutine is to be a library subroutine.

For other information on this subroutine see also the General Programming Manual (AGl) published by MDAC for professional programmers and also System Bulletin, Series 1108-10, dated 28 Oct. 1968.

Use:

Calling sequence is CALL TABIN (A, I, D, J, T, K, M, IPC, NUCASE, IEØD) where

A — is a one-dimensional array allotted for the storage of alphanumeric information. Blanks will be placed in this array before the data are input.

I — is an integer constant or variable whose value ($\geq$ 1) is the size of the array A.

D — is a one-dimensional array allotted for the storage of Table C data. Zeros will be placed in this array before the data are input.

J — is an integer constant or variable whose value ($\geq$ 1) is the size of the array D.

T — is a one-dimensional array allotted for the storage of the table of tables and the table data. Zeros will be placed in this array before the data are input.

K — is an integer constant or variable whose value ($\geq$ 1) is the size of the array T.

M — is a one-dimensional array which contains four locations. The basic deck number will be stored in M(1), the reference run number in M(2), the case number in M(3), and the part case number in M(4). Initially, this array must contain zeros. Once TABIN has been called, this array must not be changed or zeroed out.

IPC — is an integer constant or variable whose value is the part case number requested.

NUCASE is an integer constant or variable whose value is zero if the current case is to be used, one if a new case is requested.

IEØD is an integer variable whose value will be set to plus one if there is no more data, and minus one if the part case requested was not input; otherwise, it is set to zero.

Input:

TABIN reads tape NTAPE, which is set to equal FORTRAN logical tape 1.

The format of NTAPE is as follows:

Record n, n=1,3,5,...,k
Basic deck number
Reference run number
Case number
Part case number
Number of words of alphanumeric information
Number of words in Table 0
Number of words in tables other than Table 0 (including table of tables)

Record n+1, n=1,3,5,...,k
Alphanumeric information
Table 0 data
Table data other than Table 0 (including table of tables)

Record k+2
Basic deck number = 9999
Six additional words, corresponding to the format of record n.

End of file.

Error Notes:

"BD = XXX, RR = XX, CASE = XXX, PC = XX
ALPHANUMERIC ARRAY SIZE TOO SMALL = XXXXX, NECESSARY SIZE = XXXXX
EXECUTION TERMINATED BY TABIN."

"BD = XXX, RR = XX, CASE = XXX, PC = XX
ARRAY FOR TABLE 0 TOO SMALL = XXXXX, NECESSARY SIZE = XXXXX
EXECUTION TERMINATED BY TABIN."

"BD = XXX, RR = XX, CASE = XXX, PC = XX
ARRAY FOR TABLES TOO SMALL = XXXXX, NECESSARY SIZE = XXXXX
EXECUTION TERMINATED BY TABIN."

Definition of Symbols:

A is a one-dimensional array alloted for the storage of alphanumeric information.

BLANK is a word containing BCD blanks.

D is a one-dimensional array allotted for the storage of Table 0 data.

DATA is temporary storage.

I is an integer constant or variable whose value is the size of array A.

IBD is the previous basic deck number.

IBD1 is the basic deck number from tape.

ICS is the previous case number.

ICS1 is the case number from tape.

IEØD is an integer variable whose value will be set to +1 if there is no more data, and -1 if the part case requested was not input: otherwise it is set to 0.

IPC is the part case number requested.

IPC1 is the part case number from tape.

IRR is the previous reference run number.

IRR1 is the reference run number from tape.

J is an integer variable whose value is the size of array D.

K is an integer variable whose value is the size of array T.

L is used as in index.

M is a one-dimensional array which contains four locations. The basic deck number is in M(1), the reference run number is in M(2), the case number is in M(3), and the part case number is in M(4).

NALPHA is the number of words of alphanumeric information.

NTAB is the number of words in tables other than Table 0 (including table of tables).

NTAB0 is the number of words in Table 0.

NTAPE is the tape from which data are read.

NUCASE is 0 if the current case is to be used, 1 if a new case is requested.

T is a one-dimensional array allotted for the storage of the table of tables and the table data.

TACMI-4

GLOSSARY FOR SUBROUTINE TABIN 1/27/71

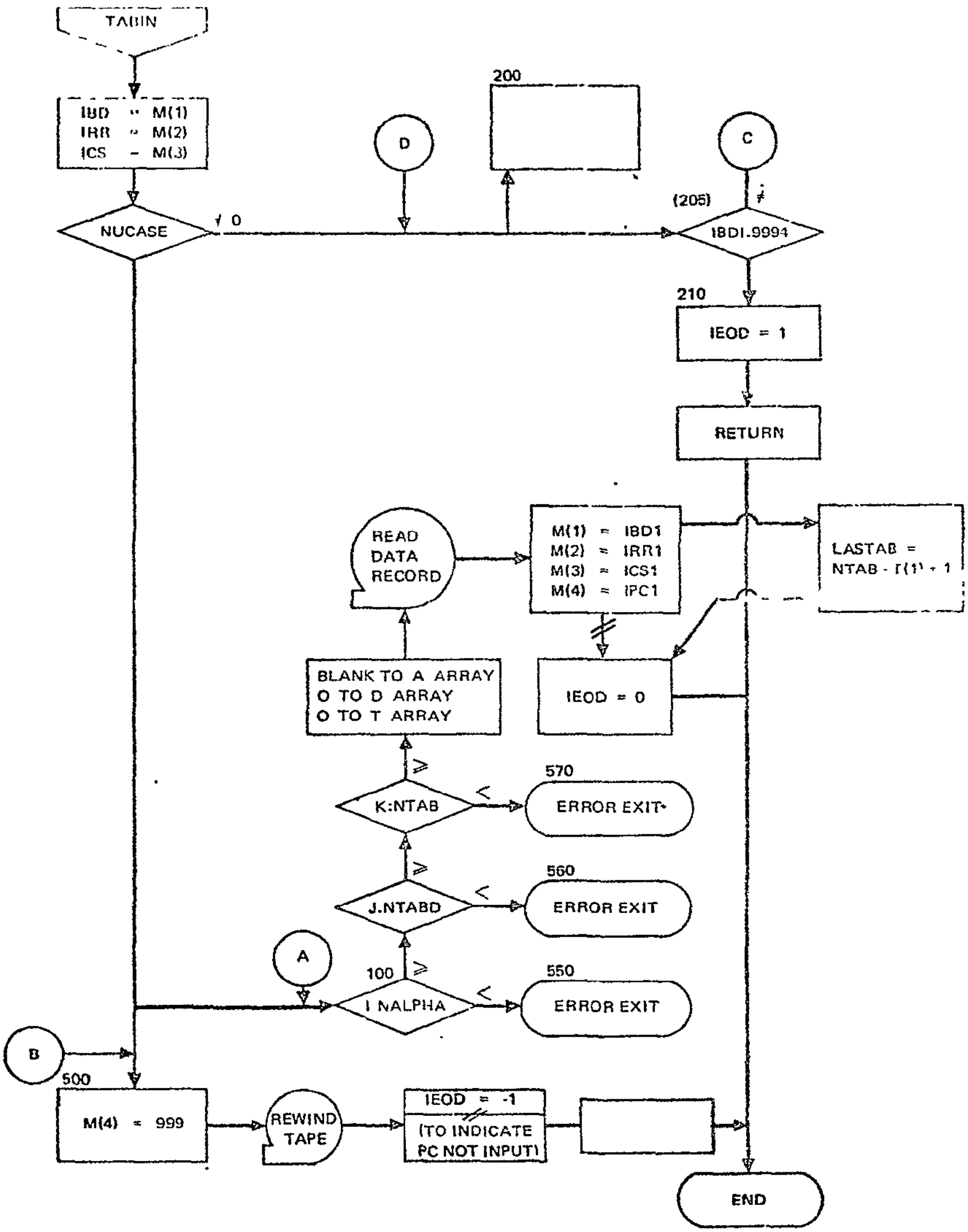
FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	COMMON SYMBOL	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	⟨A⟩	O	ONE DIMENSIONAL ARRAY ALLOTTED FOR THE STORAGE OF ALPHANUMERIC INFORMATION. (DIMENSIONLESS)	A	/TABIN /(*	)	TABIN	O	A
BLANK	⟨BLANK⟩	C	WORKING VARIABLE. (DIMENSIONLESS)	BLANK	/TABIN /(*	)	TABIN	C	BLANK
D	⟨D⟩	O	A VARIABLE LENGTH ARRAY ALLOCATED FOR THE STORAGE OF TABLE 0 (I.E. THE A ARRAY) DATA.	D	/TABIN /(*	)	TABIN	O	D
I	⟨I⟩	N	THE SIZE OF THE A ARRAY.	I	/TABIN /(*	)	TABIN	N	I
IBD	⟨IBD⟩	W	THE CURRENT BASIC DECK NUMBER	IBD	/TABIN /(*	)	TABIN	W	IBD
IBD1	⟨IBD1⟩	I	THE PREVIOUS BASIC DECK NUMBER	IBD1	/TABIN /(*	)	TABIN	I	IBD1
ICS	⟨ICS⟩	W	THE CURRENT CASE NUMBER	ICS	/TABIN /(*	)	TABIN	W	ICS
ICS1	⟨ICS1⟩	I	THE PREVIOUS CASE NUMBER	ICS1	/TABIN /(*	)	TABIN	I	ICS1
ID	⟨ID⟩	W	ARRAY(7) CASE IDENTIFICATION AND DATA SIZE INFORMATION (I E. BASIC DECK, REFERENCE RUN, CASE, PART CASE, NALPHA, NTAB, AND NTAB0)	ID	/TABIN /(*	)	TABIN	W	ID
IEOD	⟨IEOD⟩	O	THE DATA STATUS FLAG. = 1 IF THERE IS NO MORE DATA = -1 IF THE REQUESTED PART CASE WAS NOT INPUT = 0 OTHERWISE	IEOD	/TABIN /(*	)	TABIN	O	IEOD
IPC	⟨IPC⟩	I	THE PART CASE NUMBER OF THE DATA REQUESTED.	IPC	/TABIN /(*	)	TABIN	I	IPC
IPCT	⟨IPCT⟩	W	ARRAY(22) IT CONTAINS THE PART CASE NUMBERS IN THE SEQUENCE IN WHICH THEY ARE PROCESSED FOR EACH CASE.	IPCT	/TABIN /(*	)	TABIN	W	IPCT
IPC1	⟨IPC1⟩	I	THE PREVIOUS PART CASE NUMBER	IPC1	/TABIN /(*	)	TABIN	I	IPC1
IPPNTR	⟨IPPNTR⟩	W	A POINTER IN THE IPCT ARRAY USED TO PROCESS THE PROPER PART EACH FOR EACH CASE.	IPPNTR	/TABIN /(*	)	TABIN	W	IPPNTR
IRR	⟨IRR⟩	W	THE CURRENT REFERENCE RUN NUMBER	IRR	/TABIN /(*	)	TABIN	W	IRR
IRR1	⟨IRR1⟩	I	THE PREVIOUS REFERENCE RUN NUMBER	IRR1	/TABIN /(*	)	TABIN	I	IRR1
J	⟨J⟩	I	THE SIZE OF THE D ARRAY.	J	/TABIN /(*	)	TABIN	I	J
K	⟨K⟩	I	THE SIZE OF THE T ARRAY.	K	/TABIN /(*	)	TABIN	I	K
L		W	NO DESCRIPTION AVAILABLE	L	/TABIN /(*	)	TABIN	W	L
LASTAB	⟨LASTAB⟩	O	LARGEST TABLE NUMBER INPUT FOR A GIVEN CASE. (DIMENSIONLESS)	LASTAB	/LASTAB/(	1)	TABIN	O	LASTAB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	COMMON SYMBOL	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
LNTBTB	<LNTBTB>	I	THE LENGTH OF THE TABLE OF TABLES SEE THE DEFINITION OF TABLE FOR MORE DETAILED INFORMATION.	LNTBTB	/TABIN /	(*)	TABIN	I	LNTBTB
M	<M>	M	ARRAY(4). THE CASE IDENTIFICATION DATA. (BASIC DECK, REFERENCE RUN, CASE, AND PART CASE NUMBERS)	M	/TABIN /	(*)	TABIN	M	M
MAXPCT	<MAXPCT>	C	THE MAXIMUM NUMBER OF PART CASES THAT MAY BE INPUT PER CASE	MAXPCT	/TABIN /	(*)	TABIN	C	MAXPCT
MSWTCH	<MSWTCH>	W	WORKING FLAG USED TO TRANSFER TO THE CORRECT ADDRESS.	MSWTCH	/TABIN /	(*)	TABIN	W	MSWTCH
NALPHA	<NALPHA>	I	THE NUMBER OF WORDS OF ALPHANUMERIC INFORMATION.	NALPHA	/TABIN /	(*)	TABIN	I	NALPHA
NTAB	<NTAB>	I	THE NUMBER OF WORDS IN ALL TABLES OTHER THAN TABLE 0 (I.E. THE A ARRAY).	NTAB	/TABIN /	(*)	TABIN	I	NTAB
NTAB0	<NTAB0>	I	THE NUMBER OF WORDS IN TABLE 0.	NTAB0	/TABIN /	(*)	TABIN	I	NTAB0
NTAPE	<NTAPE>	C	THE FORTRAN LOGICAL FILE ON WHICH THE INPUT DATA IS WRITTEN.	NTAPE	/TABIN /	(*)	TABIN	C	NTAPE
NUCASE	<NUCASE>	I	THE NEW CASE FLAG. = 0 IF THE PRESENT CASE IS REQUESTED = 1 IF A NEW CASE IS REQUESTED	NUCASE	/TABIN /	(*)	TABIN	I	NUCASE
T	<T>	M	A VARIABLE LENGTH ARRAY ALLOCATED FOR THE STORAGE OF TABLE DATA.	T	/TABIN /	(*)	TABIN	M	T
TABIN		E	CASE OR PART CASE DATA INPUT ROUTINE.	TABIN	/TABIN /	($)	PASS1 TABIN	S E	TABIN TABIN
TBTBLN	<TBTBLN>	O	THE FLOATING POINT VALUE (REAL) OF LNTBTB.	TBTBLN	/TABIN /	(*)	TABIN	O	TBTBLN
UNUSED	<UNUSED>	O	STORAGE ALLOCATED TO READ IN UNUSED DATA (E.G. FOR BYPASSING A PART CASE OF DATA)	UNUSED	/TABIN /	(*)	TABIN	O	UNUSED

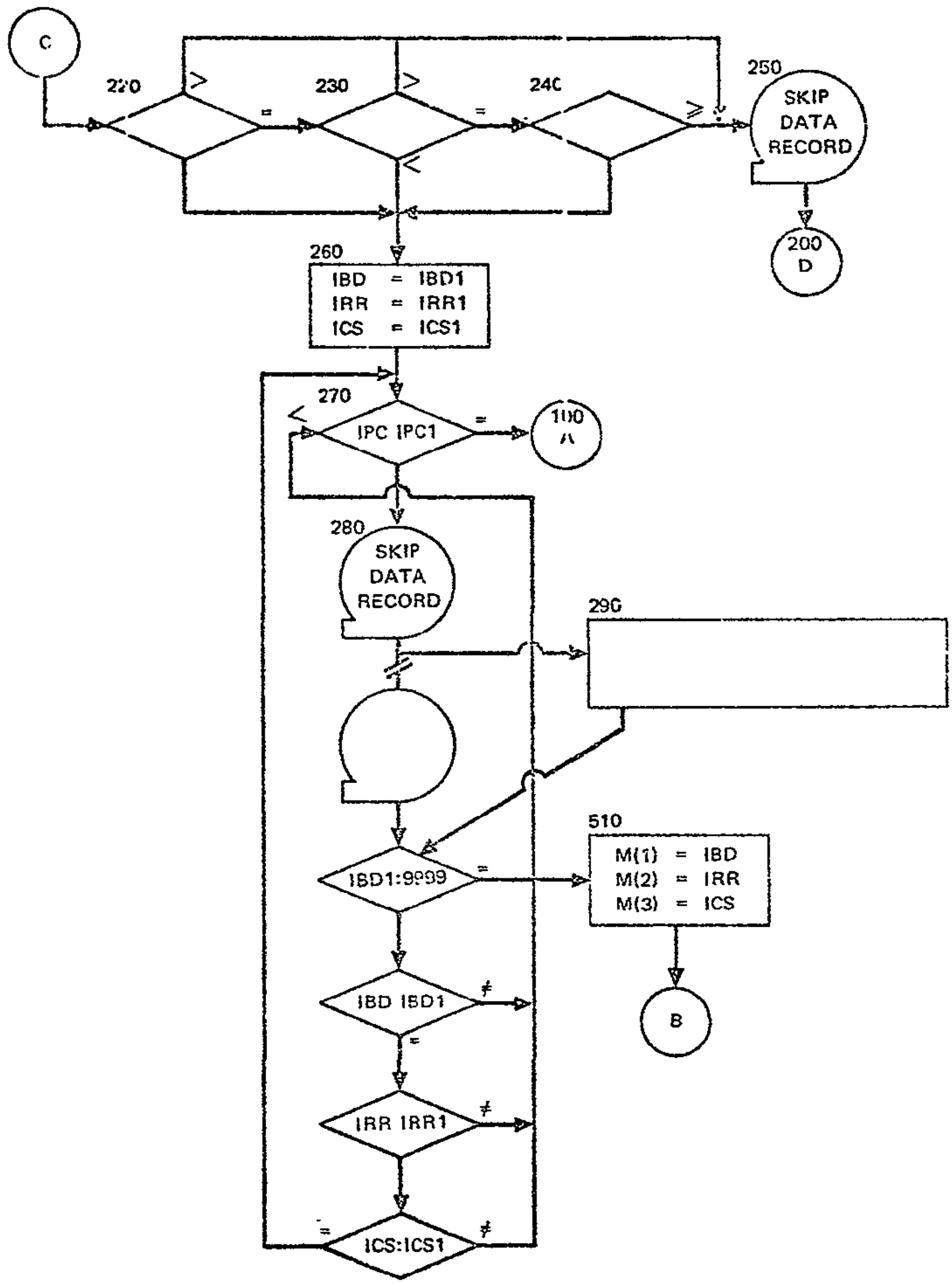
TABIN-6

2. FLOW DIAGRAM AND EQUATIONS IN ORDER OF SOLUTION

2.1 FLOW DIAGRAM

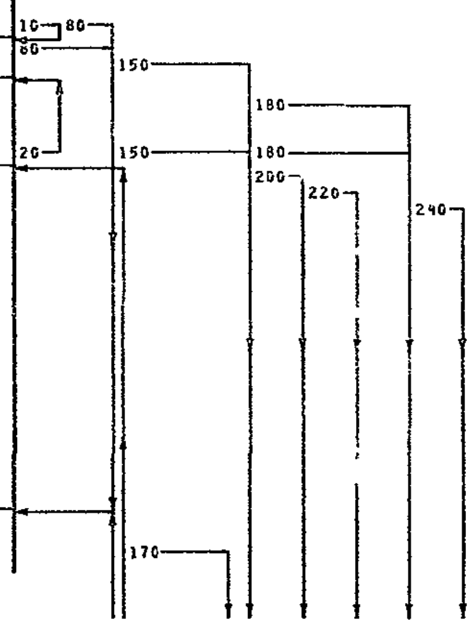


TABIN - X


C
220
230
240
250
SKIP DATA RECORD
200 D
260
IBD = IBD1
IRR = IRR1
ICS = ICS1
270
IPC IPC1
100 A
280
SKIP DATA RECORD
290
IBD1:9999
510
M(1) = IBD
M(2) = IRR
M(3) = ICS
B
IBD IBD1
IRR IRR1
ICS:ICS1

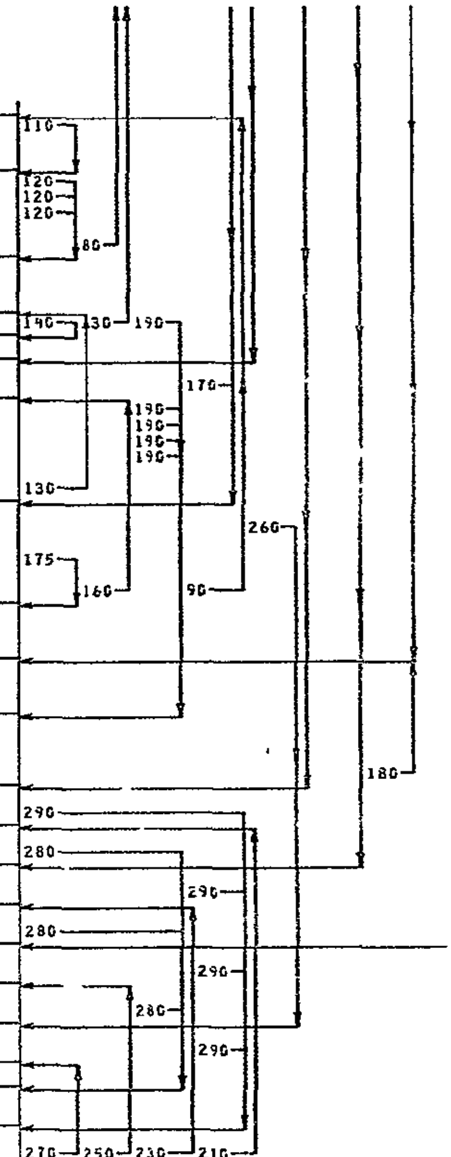
TAZIN-9

```
      SUBROUTINE TABIN (A,I,D,J,T,K,M,IPC,NUCASE,IEOD)                  TABIN
C     FORTRAN IV VARIABLE LENGTH TABLE INPUT EDITOR                     TABIN
C                                                                       TABIN
C     THIS SUBROUTINE MUST BE CALLED TO INPUT A NEW CASE OF DATA, OR TO TABIN
C     INPUT DATA FOR A DIFFERENT PART CASE OF THE CURRENT CASE.         TABIN
C                                                                       TABIN
C     CALLING SEQUENCE......CALL TABIN (A,I,D,J,T,K,M,IPC,NUCASE,IEOD)  TABIN
C     WHERE A      IS A ONE DIMENSIONAL ARRAY ALLOTTED FOR THE STORAGE  TABIN
C                  OF ALPHANUMERIC INFORMATION.                         TABIN
C           I      IS AN INTEGER CONSTANT OR VARIABLE WHOSE VALUE IS THE TABIN
C                  SIZE OF THE ARRAY A.                                 TABIN
C           D      IS A ONE DIMENSIONAL ARRAY ALLOTTED FOR THE STORAGE  TABIN
C                  OF TABLE D DATA.                                     TABIN
C           J      IS AN INTEGER CONSTANT OR VARIABLE WHOSE VALUE IS THE TABIN
C                  SIZE OF ARRAY D.                                     TABIN
C           T      IS A ONE DIMENSIONAL ARRAY ALLOTTED FOR THE STORAGE  TABIN
C                  OF THE TABLE OF TABLES AND THE TABLE DATA.           TABIN
C           K      IS AN INTEGER CONSTANT OR VARIABLE WHOSE VALUE IS THE TABIN
C                  SIZE OF ARRAY T.                                     TABIN
C           M      IS A ONE DIMENSIONAL ARRAY WHICH CONTAINS FOUR       TABIN
C                  LOCATIONS.  THE BASIC DECK NUMBER WILL BE STORED IN  TABIN
C                  M(1), THE REFERENCE RJN NUMBER IN M(2), THE CASE     TABIN
C                  NUMBER IN M(3), AND THE PART CASE NUMBER IN M(4).    TABIN
C           IPC    IS AN INTEGER CONSTANT OR VARIABLE WHOSE VALUE IS THE TABIN
C                  PART CASE NUMBER REQUESTED.                          TABIN
C           NUCASE IS AN INTEGER CONSTANT OR VARIABLE WHOSE VALUE IS    TABIN
C                  ZERO IF THE CURRENT CASE IS TO BE USED, ONE IF A NEW TABIN
C                  CASE IS REQUESTED.                                   TABIN
C           IEOD   IS AN INTEGER VARIABLE WHOSE VALUE WILL BE SET TO    TABIN
C                  PLUS ONE IF THERE IS NO MORE DATA, MINUS ONE IF THE  TABIN
C                  PART CASE REQUESTED WAS NOT INPUT, ZERO OTHERWISE.   TABIN
C                                                                       TABIN
      DIMENSION A(I),D(J),T(K),M(4)                                     TABIN
     *  , IPCT(33), ID(7)                                               AAA
      COMMON /LASTAB/ LASTAB                                            TABIN
     *,IPCT      , IPPNTR    ,NTAPE    ,MAXPCT   , BLANK                ID
      EQUIVALENCE  (IBD1,ID(1)),(IRR1,ID(2)), (ICS1,ID(3)), (IPC1,ID(4)) TABIN
     1 , (NALPHA,ID(5)), (NTABD,ID(6)), (NTAB,ID(7))  , (LNTBTB,TBTBLN)  TABIN
      IBD=M(1)                                                          TABIN
      IRR=M(2)                                                          TABIN
      ICS=M(3)                                                          TABIN
      IF (NUCASE) 80,10,80                                              TABIN
   10 IF (IPPNTR .EQ. 0)  GO TO 80                                      TABIN
      IF (IPC .GT. IPCT(IPPNTR))  GO TO 150                             TABIN
   20 IPPNTR = IPPNTR-1                                                 TABIN
      IF (IPPNTR .LT. 0) GO TO 180                                      TABIN
      BACKSPACE  NTAPE                                                  TABIN
      BACKSPACE  NTAPE                                                  TABIN
      IF (IPCT(IPPNTR+1)-IPC) 180,150,20                                TABIN
   30 IF (I LT.NALPHA) GO TO 200                                        TABIN
      IF (J LT.NTABD) GO TO 220                                         TABIN
      IF (K.LT.NTAB) GO TO 240                                          TABIN
      DO 40 L=1,I                                                       TABIN
   40 A(L)=BLANK                                                        TABIN
      DO 50 L=1,J                                                       TABIN
   50 D(L)=0.                                                           TABIN
      DO 60 L=1,K                                                       TABIN
   60 T(L)=0.                                                           TABIN
      READ (NTAPE) (A(L),L=1,NALPHA),(D(L),L=1,NTABD),(T(L),L=1,NTAB)   TABIN
      M(1)=IBD1                                                         TABIN
      M(2)=IRR1                                                         TABIN
      M(3)=ICS1                                                         TABIN
      M(4)=IPC1                                                         TABIN
      TBTBLN = T(1)                                                     TABIN
      LASTAB = NTAB-LNTBTB+1                                            TABIN
      IEOD = 0                                                          TABIN
   70 RETURN                                                            TABIN
C                                                                       TABIN
C     FIND NEXT CASE                                                    TABIN
C                                                                       TABIN
   80 IPPNTR = 0                                                        TABIN
      ASSIGN 90 TO MSWTCH                                               TABIN
      GO TO 170                                                         TABIN
```



97

```
      90 IF (IBD1 .NE. 9999) GO TO 110                                    TABIN
     100 IEOD=1                                                           TABIN
         RETURN                                                           TABIN
     110 IF (ICS1 .NE  ICS)  GO TO 120                                    TABIN
         IF (IRR1 .NE. IRR)  GO TO 120                                    TABIN
         IF (IBD1 .NE. IBD)  GO TO 120                                    TABIN
         READ (NTAPE)  UNUSED                                             TABIN
         GO TO 80                                                         TABIN
     120 IBD=IBD1                                                         TABIN
         IRR=IRR1                                                         TABIN
         ICS=ICS1                                                         TABIN
     130 IF (IPC-IPC1) 190,30,140                                         TABIN
     140 READ (NTAPE)  UNUSED                                             TABIN
     150 ASSIGN 160 TO MSWTCH                                             TABIN
         GO TO 170                                                        TABIN
     160 IF (IBD1 .EQ  9999) GO TO 190                                    TABIN
         IF (IBD.NE IBD1) GO TO 190                                       TABIN
         IF (IRR.NE IRR1) GO TO 190                                       TABIN
         IF (ICS.NE.ICS1) GO TO 190                                       TABIN
   C                                                                      TABIN
         GO TO 130                                                        TABIN
     170 IPPNTR = IPPNTR+1                                                TABIN
         IF (IPPNTR  GT  MAXPCT)  GO TO 260                               TABIN
         READ (NTAPE) ID                                                  TABIN
         IF(IPCT(IPPNTR) GT IPC AND.NUCASE.EQ.0) GO TO 175                SEP18
         IPCT(IPPNTR) = IPC1                                              TABIN
         GO TO MSWTCH, (90,160)                                           TABIN
     175 READ (NTAPE)                                                     SEP18
         GO TO MSWTCH                                                     SEP18
   C                                                                      TABIN
     180 M(4)=999                                                         TABIN
         IEOD=-1                                                          TABIN
         RETURN                                                           TABIN
     190 M(1)=IBD                                                         TABIN
         M(2)=IRR                                                         TABIN
         M(3)=ICS                                                         TABIN
         GO TO 180                                                        TABIN
     200 ASSIGN 210 TO MSW                                                TABIN
         GO TO 290                                                        TABIN
     210 WRITE (6,300) I,NALPHA                                           TABIN
         GO TO 280                                                        TABIN
     220 ASSIGN 230 TO MSW                                                TABIN
         GO TO 290                                                        TABIN
     230 WRITE (6,310) J, NTABO                                           TABIN
         GO TO 280                                                        TABIN
     240 ASSIGN 250 TO MSW                                                TABIN
         GO TO 290                                                        TABIN
     250 WRITE (6,320) K, NTAB                                            TABIN
         GO TO 280                                                        TABIN
     260 ASSIGN 270 TO MSW                                                TABIN
         GO TO 290                                                        TABIN
     270 WRITE (6,340)  MAXPCT                                            TABIN
     280 WRITE (6,350)                                                    TABIN
         STOP                                                             TABIN
     290 WRITE (6,330) (ID(I),I=1,4)                                      TABIN
         GO TO MSW, (210,230,250,270)                                     TABIN
     300 FORMAT (1H ,34HALPHANUMERIC ARRAY SIZE TOO SMALL=I5,17H, NECESSARY TABIN
        1 SIZE=I5)                                                        TABIN
     310 FORMAT (1H ,28HARRAY FOR TABLE 0 TOO SMALL=I5,17H, NECESSARY SIZE= TABIN
        1I5)                                                              TABIN
     320 FORMAT (1H ,27HARRAY FOR TABLES TOO SMALL=I5,17H, NECESSARY SIZE=  TABIN
        1I5)                                                              TABIN
     330 FORMAT (1H1,3HBD=I3,5H, RR=I2,7H, CASE=I3,5H, PC=I2)             TABIN
     340 FORMAT (10H MORE THAN,I4,10H PC'S/CASE)                          TABIN
     350 FORMAT (1H0,30HEXECUTION TERMINATED BY TABIN.)                   TABIN
```



20 OCT 72 G 01-46

139.   END   TABIN

20 OCT 72 6.01-46

99

SUBRØUTINE
WTDRP

100

Subroutine WTDRP

Purpose

Subroutine WTDRP calculates the weight dropped at booster staging in the rubber-stage optimization problem. It also calculates the sensitivities of the weight drop with respect to the booster burn-out weight.

Description

The equations for this routine are described in Section 13 of Volume I.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
BK1		I	Value of constant weight in booster stage weight equation	/SIZING/(	279)	SIZ1	I	BK1
						SIZ2	I	BK1
						WTDRP	I	BK1
BK2		I	Value of linear term coefficient in booster stage weight equation	/SIZING/(	280)	SIZ1	I	BK2
						SIZ2	I	BK2
						WTDRP	I	BK2
BK3		I	Value of 1/3-power term coefficient in booster stage weight equation	/SIZING/(	281)	SIZ1	I	BK3
						SIZ2	I	BK3
						WTDRP	I	BK3
BK4		I	Value of 2/3-power term coefficient in booster stage weight equation	/SIZING/(	282)	SIZ1	I	BK4
						SIZ2	I	BK4
						WTDRP	I	BK4
DWEB		M	Sensitivity of booster stage weight to propellant weight (lb/lb)	/SIZING/(	274)	PAYLOD	I	DWEB
						SIZOUT	I	DWEB
						SIZ1	M	DWEB
						SIZ2	M	DWEB
						STAU	I	DWEB
						WTDRP	M	DWEB
DWEO		M	Sensitivity of orbiter stage weight to propellant weight (lb/lb)	/SIZING/(	275)	PAYLOD	I	DWEO
						SIZOUT	I	DWEO
						SIZ1	M	DWEO
						SIZ2	M	DWEO
						SIZ4	M	DWEO
						WTDRP	M	DWEO
ITNBW		I	Booster empty weght curve no.	/SIZING/(	317)	SIZE	O	ITNBW
						SIZ1	I	ITNBW
						SIZ2	I	ITNBW
						WTDRP	I	ITNBW
ITNOW		I	Orbiter empty weight curve no.	/SIZING/(	318)	SIZE	O	ITNOW
						SIZ1	I	ITNOW
						SIZ2	I	ITNOW
						SIZ4	I	ITNOW
						WTDRP	I	ITNOW
OK1		I	Same as bk1 except for orbiter	/SIZING/(	286)	SIZ1	I	OK1
						SIZ2	I	OK1
						SIZ4	I	OK1
						WTDRP	I	OK1
OK2		I	Same as bk2 except for orbiter	/SIZING/(	287)	SIZ1	I	OK2
						SIZ2	I	OK2
						SIZ4	I	OK2
						WTDRP	I	OK2
OK3		I	Same as bk3 except for orbiter	/SIZING/(	288)	SIZ1	I	OK3
						SIZ2	I	OK3
						SIZ4	I	OK3
						WTDRP	I	OK3
OK4		I	Same as bk4 except for orbiter	/SIZING/(	289)	SIZ1	I	OK4
						SIZ2	I	OK4
						SIZ4	I	OK4
						WTDRP	I	OK4
WEB		M	Booster stage weight (lb)	/SIZING/(	304)	PAYLOD	I	WEB
						SIZOUT	I	WEB
						SIZ1	M	WEB
						SIZ2	M	WEB
						SIZ4	I	WEB
						TAMPAR	I	WEB
						WTDRP	M	WEB
WEO		M	Orbiter stage weight (lb)	/SIZING/(	303)	PAYLOD	I	WEO
						SIZOUT	I	WEO
						SIZ1	M	WEO
						SIZ2	M	WEO
						SIZ3	I	WEO
						SIZ4	M	WEO
						TAMPAR	I	WEO
						WTDRP	M	WEO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
WPB		M	Booster propellant weight (lb)	/SIZING/(	277)	SIZOUT	I	WPB
						SIZ1	M	WPB
						SIZ2	M	WPB
						SIZ4	I	WPB
						TAMPAR	I	WPB
						WTDRP	M	WPB
WPO		M	Orbiter propellant weight (lb)	/SIZING/(	312)	SIZOUT	I	WPO
						SIZ1	M	WPO
						SIZ2	M	WPO
						SIZ3	I	WPO
						SIZ4	M	WPO
						TAMPAR	I	WPO
						WTDRP	M	WPO

```
      SUBROUTINE WTDRP(WPP,W1D,W1DW,I)                                  WTDRP
C       THIS  ROUTINE  CALCULATES  WEIGHT  DROPPED  FOR OPTIMAL STAGING  WTDRP
C       AND    PARTIAL OF WT.DROPPED WITH RESPECT  TO  BOOSTER CUT-OFF W WTDRP
      REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                           SIZING
      COMMON /SIZING/                                                   SIZING
C            PHASE II SIZING PARAMERERS                                 SIZING
     *TZ,      VV(3),     QP(14),   EROR,    PZ(5),     VQ,      SW(20), SIZING
     *SV(28),  SQ(37,5), SE(11),   TLAT,     TLNG,                       SIZING
C          PHASE I SIZING PARAMERERS                                    SIZING
     *WBO,     WLOO,      DWEB,     DWEO,     TOLWT,     WPB,     TWRAT2, SIZING
     *BK1,     BK2,       BK3,      BK4,      ISIZE,     TRAFLG,  TWRATO, SIZING
     *OK1,     OK2,       OK3,      OK4,      PRFLG,     IPASS,   IPSMAX, SIZING
     *AEXIT,   TVACO,     NO,       WFO,      IDVEL,     ISPO,    ISPB,   SIZING
     *XPL,     TVACB,    NNB,       WEO,      WEB,       WO,      WLO,    SIZING
     *DVO,      DVB,       MUB,      MUO,      VSTG,      WPO             SIZING
     *,JTYP,    BECO   ,  BSTG   ,  ORBI,   ITNBW  , ITNOW   ,            SIZING
     * SVOPSQ   , SVOCON   ,IHUNT   ,IOPSTG  ,ISZD(19)                   UH
      IF( I.LT.0 ) GO TO 3                                              JULY28
      WPB = WPP                                                         JULY28
C                                                                       JULY28
C     BOOSTER STAGE WT AND SENSITIVITY                                  JULY28
C                                                                       JULY28
    1 IF(BK1.GT.0.0) GO TO 2                                            JULY28
      CALL SPLIZ( ITNBW, WPB, WEB, DWEB)                                JULY28
      GO TO 5                                                           JULY28
    2 WEB = BK1 + BK2* WPB+ BK3* WPB**0.3333+ BK4* WPB**0.6667          JULY28
      DWEB= BK2 + BK3* 0.3333* WPB**(-0.6667)+ BK4* 0.6667* WPB**       JULY28
     *        (-0.3333)                                                 JULY28
    5 W1D = WEB                                                         JULY28
      W1DW = DWEB                                                       JULY28
      RETURN                                                            JULY28
C                                                                       JULY28
C     ORBITER STAGE WT AND SENSITIVITY                                  JULY28
C                                                                       JULY28
    3 WPO = WPP                                                         JULY28
      IF(OK1.GT.0.0)  GO TO 4                                           JULY28
      CALL SPLIZ(ITNOW, WPO, WEO, DWEO)                                 JULY28
      GO TO 6                                                           JULY28
    4 WEO = OK1+ OK2*WPO+ OK3* WPO** 0.3333+ OK4* WPO** 0.6667          JULY28
      DWEO= OK2+ 0.3333* OK3* WPO**(-0.6667)+ 0.6667* OK4* WPO**        JULY28
     *        (-0.3333)                                                 JULY28
    6 W1D = WEO                                                         JULY28
      W1DW = DWEO                                                       JULY28
      RETURN                                                            JULY28
      END                                                               JULY28
```

Input Editor (INEDIT) 3

CONTENTS

TABE

TABE

Purpose

TABE is a dummy executive program that heads up the input edit overlay (overlay containing INEDIT).

```
1          PROGRAM TABE                                          TABE
2          CALL INEDIT                                           TABE
3          RETURN                                                TABE
4.         END                                                   TABE
```


111 4

SUBRØUTINE
INEDIT

Subroutine INEDIT

Purpose

INEDIT reads input data card images using NAMELIST and stores cases and part cases on input file 1 for access by the various modules of PADS.

Description

INEDIT first reads each data card separately, prints its image, and then copies it on to File 4. File 4 will then contain all data card images, and serve as the input file for subsequent NAMELIST reads.

The first NAMELIST read ($XX) either reads in control flags or fills common block /AA/ with trajectory and Phase I sizing data.

At the end of common block /AA/ are three arrays used for storing the data from the other NAMELIST reads associated with SSSP sizing problems. These NAMELIST sets are $DATA 3 and $DATA 2.

After reading data into common block/AA/, INEDIT scans for key quantities that have actually been input (words that are non-negative zero) and sets up part cases to be stored on file 1.

The part case and table structure resulting from this processing are charted below.

Part Case No.	Description of Data	Common block/ or (routine) where date used	Extent In /AA/ array
1	fixed length table Tables 1→30 variable length univariant data	/GLOBAL/ /TBLE/ (SPLICO)	GR → DUM 5 CLA (70, 30)
2	Table 1 Initial conditions Table 2 Target condition	(GEINP) (BNDRYC)	T1→TIME1 PAYOFF-CT20

Part Case No.	Description of Data	Common block/ or (routine) where date used	Extent In AA array
4	Table 1, steepest descent starting solution phase sequencing	(SDINP)	PH1→PH20
5	Table O, steepest descent convergence data	/STS/ (SDINP)	DPAY→DLPI
	Table 1-20 steepest descent starting solution control history		TCØNA
6	Table O Bivariate table arguments.	(BLICØ)	TALFA1→ TMACH1
	Tables 1-31 Bivariate table.		TCLCD1
7	Table O, Bivariate table arguments. Table 1-31 Bivariate table.	(BLICØ)	TALFA2→ TMACH2 TCLCD2
8	Table O, Bivariate table arguments.	(BLICØ)	TALFA3→ TMACH3
	Tables 1-31 Bivariate table.		TCLCD3
9	Table O, Bivariate table arguments	(BLICØ)	TVELT→ TALT
	Tables 1-31 Bivariate table.		THSF
10	Table O sizing data	/SIZING/	WBØ→ISZD
11→30	Table O ARC data	/ARCDAT/	SREF→ DUM20
31	Table O Booster data		AB
32	Table O Orbiter data		AC
33	Synthesis data		AD

```
      SUBROUTINE INEDIT                                                 INEDIT
      COMMON/TT/INTBD,INTRR,INTCS,IPC,NALPHA,NTABO,NTAB,NT,NTOT,JTOT(31) INEDIT
      EQUIVALENCE (IDENT,INTBD),(ITOT,NTOT)                             INEDIT
      DIMENSION IDENT(7),ITOT(32)                                       INEDIT
      COMMON/FLAGS/ KMOD,KMAKBD,BD,RR,CS,MODCS,MAKEBD,SIZING            INEDIT
C                                                                       INEDIT
      INTEGER BD,RR,CS,SIZING,EXTBD,EXTRR,EXTCS,OUT                     INEDIT
      REAL                                                              INEDIT
     1    INNER,ITRMAX,NARC,                                            INEDIT
     2    MU1,                                                          INEDIT
     5    NITER,                                                        INEDIT
     9 ISIZE,IPASS,IPSMAX,NO,IOVEL,ISPO,ISPB,NNB,MUB,JTYP,ITNBW,ITNOW,  INEDIT
     9ISZD,                                                             INEDIT
     A    IATM,IMODE,JAER,JPRO,LFTMAX                                   INEDIT
C     FOR DATA3                                                         INEDIT
      REAL ISP,ITPS,K,KIN,LF,MR,NCREW,NENGS,NLISTO,NPASS,NWL            INEDIT
C     FOR DATA4                                                         INEDIT
      REAL        ISLB,ISLO,IVACB,IVACO,NXFOB,IOVELO                    INEDIT
C                                                                       INEDIT
      DIMENSION                                                         INEDIT
     1   AA(     1),TCLO(70,30),TCDO(70,30),TFK(70,30),TCMO(70,30),     INEDIT
     1 TCMA(70,30),TBLDQ(70,30),TXCGW(70,30),TZCGW(70,30),TDBH(70,30),  INEDIT
     1 TWDRP(70,30),TFVAC(70,30),TISPL(70,30),                          INEDIT
     5 TCONB(30,20)                                                     INEDIT
      EQUIVALENCE                                                       INEDIT
     1(TCLA,TCLO,TCDO,TFK,TCMO,TCMA,TBLDQ,TXCGW,TZCGW,TDBH,TWDRP,TFVAC, INEDIT
     1 TISPL), (AA,GR),                                                 INEDIT
     5 (TCONA,TCONB),                                                   POW
     *(TNCLA,SNA),(DUM9 ,SNB)                                           FIXED
      DIMENSION SNB (20),SNA(20)                                        FIXED
C                                                                       INEDIT
      COMMON/AA/                                                        INEDIT
     1GR,ER,OMGZ,RHORF,YMURF,PRCO,DUM1,EPSLON,INNER,ITRMAX,DUM2(7),NARC, INEDIT
     1DUM3(49),PSIRF,PFLG1,PFLG2,PFLG3,PFLG4,DUM4(20),TPSOL,DUM5(9),    INEDIT
     1TCLA(70,30),                                                      INEDIT
     2T1 (2),T2 (2),T3 (2),T4 (2),T5 (2),T6 (2),T7 (2),T8 (2),T9 (2),   INEDIT
     2T10(2),T11(2),T12(2),T13(2),T14(2),T15(2),T16(2),T17(2),T18(2),   INEDIT
     2T19(2),T20(2),V1 (2),V2 (2),V3 (2),V4 (2),V5 (2),V6 (2),V7 (2),   INEDIT
     2V8 (2),V9 (2),V10(2),V11(2),V12(2),V13(2),V14(2),V15(2),V16(2),   INEDIT
     2V17(2),V18(2),V19(2),V20(2),W1 (2),W2 (2),W3 (2),W4 (2),W5 (2),   INEDIT
     2W6 (2),W7 (2),W8 (2),W9 (2),W10(2),W11(2),W12(2),W13(2),W14(2),   INEDIT
     2W15(2),W16(2),W17(2),W18(2),W19(2),W20(2),GAM1(2),ALT1(2),PSI1(2), INEDIT
     2RND1(2),MU1(2),H1(2),SQ1(2),TIM1(2),                              INEDIT
     2PAYOFF(3),CN1(3),CN2(3),CN3(3),CN4(3),CN5(3),CN6(3),CN7(3),CN8(3), INEDIT
     2CT1 (2),CT2 (2),CT3 (2),CT4 (2),CT5 (2),CT6 (2),CT7 (2),CT8 (2),  INEDIT
     2CT9 (2),CT10(2),CT11(2),CT12(2),CT13(2),CT14(2),CT15(2),CT16(2),  INEDIT
     2CT17(2),CT18(2),CT19(2),CT20(2),                                  INEDIT
     4PH1 (6),PH2 (6),PH3 (6),PH4 (6),PH5 (6),PH6 (6),PH7 (6),PH8 (6),  INEDIT
     4PH9 (6),PH10(6),PH11(6),PH12(6),PH13(6),PH14(6),PH15(6),PH16(6),  INEDIT
     4PH17(6),PH18(6),PH19(6),PH20(6)                                   INEDIT
      COMMON/AA/                                                        INEDIT
     5OPAY,PMIN,TOPEN1,TOPEN2,DUM6(7),PHIWT,DUM7(18),NITER,TOF(40),     INEDIT
     5WPT1(9),DUM8,DLP1(9),TCONA(30,20),                                INEDIT
     6TALFA1(31),TMACH1(31),TCLCD1(62,31),                              INEDIT
     7TALFA2(31),TMACH2(31),TCLCD2(62,31),                              INEDIT
     8TALFA3(31),TMACH3(31),TCLCD3(62,31),                              INEDIT
     *TVELT(31),TALT(31),THSF(62,31) ,                                  JULY28
     9WBO(1),  DUM21,    DUM22,    DUM23,    TOLWT,    WPB,    DUM24,   INEDIT
     9BK1,     BK2,      BK3,      BK4,      ISIZE,    TRAFLG, TWRATO,  INEDIT
     9OK1,     OK2,      OK3,      OK4,      PRFLG,    IPASS,  IPSMAX,  INEDIT
     9EXITA,   TVACO,    NO,       WFD,      IOVEL,    ISPO,   ISPB,    INEDIT
     9XPL,     TVACB,   NNB,       WEO,      WEB,      WO,     WLO,     INEDIT
     9DUM25,   DUM26,     MUB,     DUM27,     VSTG,     WPO,            INEDIT
     9JTYP,    BECO,     BSTG,     ORBI,     ITNBW,   ITNOW,  ISZD(13)  INEDIT
      COMMON/AA/                                                        INEDIT
     ASREF  (20),AEXIT (20),XISP  (20),TMULT (20),DTNC  (20),PFRQ  (20), INEDIT
     AIATM  (20),IMODE (20),JAER  (20),JPRO  (20),QMAX  (20),GMAX  (20), INEDIT
     ALFTMAX(20),HDMAX (20),GMDOT (20),ALFMAX(20),PHMAX (20),TNCLA (20), INEDIT
     ATNCLO (20),TNCDO (20),TNFK  (20),TNCMO (20),TNCMA (20),TNBLDQ(20), INEDIT
     ATNFVAC(20),TNISPL(20),TNXCGW(20),TNZCGW(20),TNWDRP(20),DUM9  (20), INEDIT
     ATNDBH (20),XCGR  (20),ZCGR  (20),XE    (20),ZE    (20),XT    (20), INEDIT
     ADREF  (20),DUM10 (20),DUM11 (20),HTFLG (20),REMAX (20),FRATE (20), INEDIT
     ADUM13 (20),DUM14 (20),DUM15 (20),DUM16 (20),DUM17 (20),DUM18 (20), INEDIT
     ADUM19 (20),DUM20 (20)                                             INEDIT
      EQUIVALENCE (IHUNT,ISZD(3))                                       POW
```

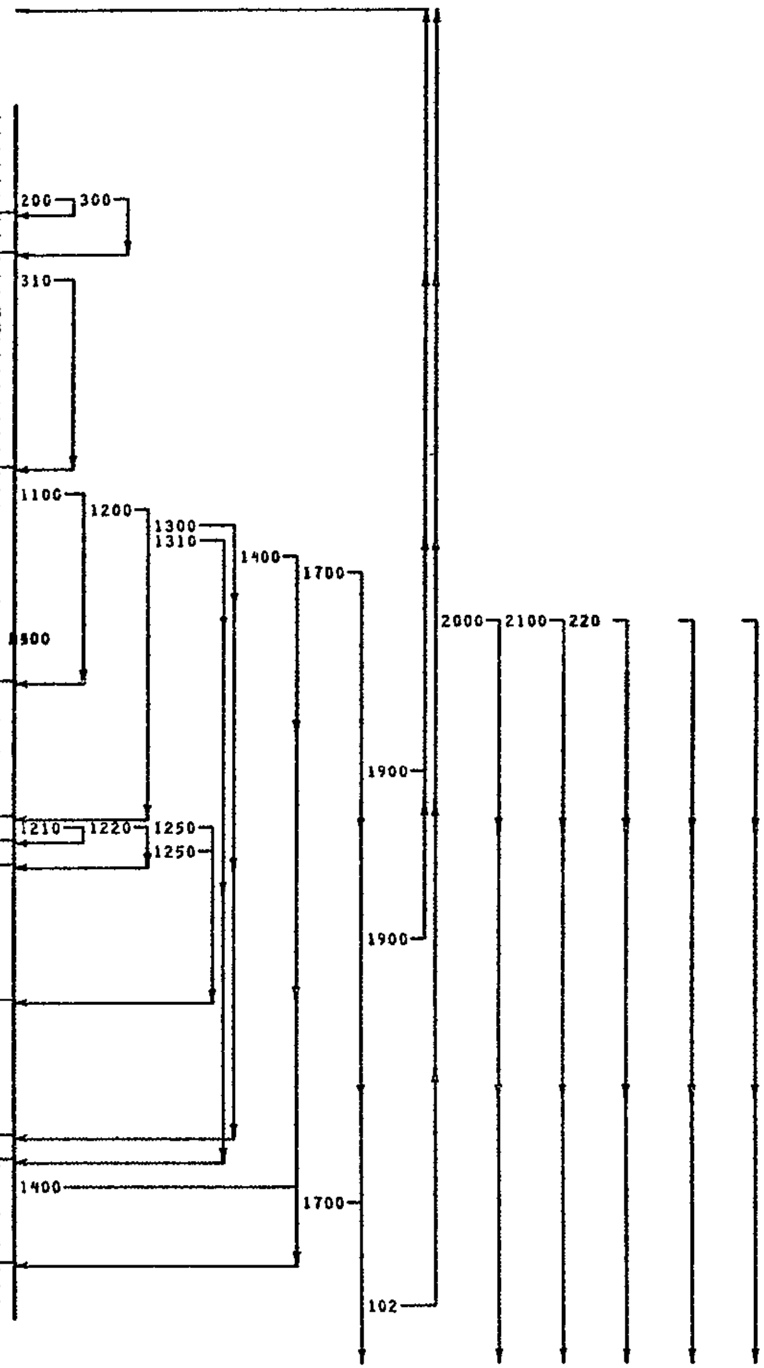
20 OCT 72 G 01-46

```
      *,(SOCO,ISZD(4)),(SOSP,ISZD(5))                                   JULY28
       REAL IHUNT                                                       POW
C                                                                       INEDIT
       NAMELIST/XX/ BD,RR,C5,MODCS,MAKEBD,SIZING,IHUNT,NODATA           POW
      1,GR,ER,OMGZ,RHORF,YMURF,PRCO,EPSLON,INNER,ITRMAX,NARC,PSIRF,     INEDIT
      2 PFLG1,PFLG2,PFLG3,PFLG4,TPSOL,TCLA,TCLO,TCDO,TFK,TCMO,TCMA,TBLOQ, INEDIT
      3 TXCGW,TZCGW,TDBH,TWDRP,TFVAC,TISPL,SOCO,SOSP,                   JULY28
      4T1,T2,T3,T4,T5,T6,T7,T8,T9,T10,T11,T12,T13,T14,T15,T16,T17,T18,T19 INEDIT
      5,T20,V1,V2,V3,V4,V5,V6,V7,V8,V9,V10,V11,V12,V13,V14,V15,V16,V17, INEDIT
      6V18,V19,V20,W1,W2,W3,W4,W5,W6,W7,W8,W9,W10,W11,W12,W13,W14,W15,W16 INEDIT
      7,W17,W18,W19,W20,GAM1,ALT1,PSI1,RHO1,MU1,H1,SO1,TIM1,PAYOFF,CN1, INEDIT
      8CN2,CN3,CN4,CN5,CN6,CN7,CN8,CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9, INEDIT
      9CT10,CT11,CT12,CT13,CT14,CT15,CT16,CT17,CT18,CT19,CT20,PH1,PH2,  INEDIT
      0PH3,PH4,PH5,PH6,PH7,PH8,PH9,PH10,PH11,PH12,PH13,PH14,PH15,PH16,  INEDIT
      1PH17,PH18,PH19,PH20,DPAY,PMIN,TOPEN1,TOPEN2,PHIWT,NITER,TOF,WPTI, INEDIT
      2OLPI,TCONA,TCONB,TALFA1,TMACH1,TCLCO1,TALFA2,TMACH2,TCLCO2,TALFA3, INEDIT
      3TMACH3,TCLCO3,SREF,AEXIT,XISP,TMULT,DINC,PFRQ,IATM,IMODE,JAER,JPRO INEDIT
      4,QMAX,GMAX,LFTMAX,HDMAX,GMDOT,ALFMAX,PHMAX,TNCLA,TNCLO,TNCDO,TNFK, INEDIT
      5 TNCMO,TNCMA,TNBLOQ,TNFVAC,TNISPL,TNXCGW,TNZCGW,TNWDRP,TNDBH,XCGR, INEDIT
      6ZCGR,XE,ZE,XT,OREF,HTFLG,REMAX,FRATE,TALT,TVELT,THSF,JTYP,SNA,SNB, FIXED
      7WBO,TOLWT,WPB,BK1,BK2,BK3,BK4,ISIZE,TRAFLG,TWRATO,OK1,OK2,OK3,OK4, INEDIT
      8PRFLG,IPASS,IPSMAX,EXITA,TVACO,NO,WFO,IDVEL,ISPO,ISPB,XPL,TVACB, INEDIT
      9NNB,WEO,WEB,WO,WLO,MUB,VSTG,WPO,BECO,BSTG,ORBI,ITNBW,ITNOW,ISZD  UH
C                                                                       INEDIT
       DIMENSION  C(300),CFUEL(6),DEF(5),ISP(6),K(30),MR(6)             INEDIT
       DIMENSION  AB(392),AC(392), AD(101)                              POW
       EQUIVALENCE (AB,AA(12212)),(AC,AA(12604)),(AD,AA(12996))         FIXED
       EQUIVALENCE                                                      INEDIT
      1(ANENGS,AB(   1)),(ANTANK,AB(   2)),(ASRATO,AB(   3)),(ASWEEP,AB(4)), INEDIT
      2(C     ,AB(   5)),(CBBODY,AB(305)),(CFUEL ,AB(306)),             INEDIT
      3(CHBODY,AB(312)),(CLBODY,AB(313)),(CSBODY,AB(314)),              INEDIT
      4(CSFAIR,AB(315)),(CSFUTK,AB(316)),(CSHORZ,AB(317)),              INEDIT
      5(CSOXTK,AB(318)),(CSPLAN,AB(319)),(CSVERT,AB(320)),              INEDIT
      6(CSWING,AB(321)),(CTHRST,AB(322)),(CTHST2,AB(323)),              INEDIT
      7(DEF   ,AB(324)),(FXWOVS,AB(329)),(ISP   ,AB(330)),              INEDIT
      8(ITPS  ,AB(336)),(K     ,AB(337)),(KIN   ,AB(367)),              INEDIT
      9(LF    ,AB(368)),(MR    ,AB(369)),(NCREW,AB(375)),               INEDIT
      0(NENGS ,AB(376)),(NLISTO,AB(377)),(NPASS ,AB(378)),              INEDIT
      1(NWL   ,AB(379)),(PCHAM ,AB(380)),(Q     ,AB(381)),              INEDIT
      2(RHOFU ,AB(382)),(RHOFU2,AB(383)),(RHOX  ,AB(384)),              INEDIT
      3(RHOX2 ,AB(385)),(SBODY ,AB(386)),(TOL   ,AB(387)),              INEDIT
      4(TOVERC,AB(388)),(TPRATO,AB(389)),(TYTAIL,AB(390)),              INEDIT
      5(VBODY ,AB(391)),(WGROSS,AB(392))                                INEDIT
       EQUIVALENCE                                                      UH
      1(ALD   ,AD(   1)),(FBPAR ,AD(   2)),(IDVELO,AD(   3)),           UH
      2(ISLB  ,AD(   4)),(ISLO  ,AD(   5)),(IVACB ,AD(   6)),           UH
      3(IVACO ,AD(   7)),(PERISP,AD(   8)),(QMXX  ,AD(   9)),           UH
      4(QMXS  ,AD(  10)),(SFC   ,AD(  11)),(SLVOUT,AD(  12)),           UH
      5(COPIES,AD(  13)),(SYNIT ,AD(  14)),(TFCTRB,AD(  15)),           UH
      6(TFCTRO,AD(  16)),(TOLMU ,AD(  17)),(TOLTW ,AD(  18)),           UH
      7(TRATIO,AD(  19)),(TWLO  ,AD(  20)),(TWLOI ,AD(  21)),           UH
      8(WTOUT ,AD(  22)),(FIRE  ,AD(  23)),(BOOTW ,AD(  24)),           UH
      9(VCRUSE,AD(  25)),(NXFOB ,AD(  26)),(PRNTX ,AD(  27)),           UH
      0(FSEC  ,AD(  28)),(CLVG  ,AD(  29)),(DRNG  ,AD(  30)),           UH
      1(SOLID ,AD(  31)),(AS    ,AD(  32)),(BS    ,AD(  33)),           UH
      2(SISP  ,AD(  34)),(SIHERT,AD(  35)),(SAE   ,AD(  36)),           UH
      3(TSBO  ,AD(  37)),(FLYBCK,AD(  38)),(WPOREQ,AD(  39)),           UH
      4(WOREQ ,AD(  40)),(GWREQ ,AD(  41)),(FBFUEL,AD(  42)),           UH
      5(CA    ,AD(  43)),(CB    ,AD(  44)),(WFLYX ,AD(  45))            UH
       EQUIVALENCE                                                      UH
      *(RT   ,AD(46)),(R1   ,AD(47)),(R3   ,AD(48)),                    UH
      *(SFC1   ,AD(49)),(SFC2   ,AD(50)),(SFC3   ,AD(51)),              UH
      *(ALD1,  AD(52)),(ALD2  ,AD(53)),(ALD3,AD(54)),                   UH
      *(VFLY1  ,AD(55)),(VFLY2 ,AD(56)),(VFLY3,AD(57))                  UH
       EQUIVALENCE                                                      POW
      *(PNDX,AD(60)),(BLOW,AD(70)),(BUPP,AD(80)),(STEP,AD(90)),         POW
      *(PAYX,AD(101)),(RVAR,AD(59))                                     POW
       DIMENSION PNDX(10),BUPP(10),BLOW(10),STEP(11)                    POW
C                                                                       INEDIT
C      NAMELIST TO  READ IN DATA                                        INEDIT
C                                                                       INEDIT
       NAMELIST/DATA3/                                                  INEDIT
      1ANENGS  ,ANTANK   ,ASRATO   ,ASWEEP   ,C        ,CBBODY   ,CFUEL    , INEDIT
      2CHBODY  ,CLBODY   ,CSBODY   ,CSFAIR   ,CSFUTK   ,CSHORZ   ,CSOXTK   , INEDIT
      3CSPLAN  ,CSVERT   ,CSWING   ,CTHRST   ,CTHST2   ,DEF      ,FXWOVS   , INEDIT
```

20 OCT 72 6.01-46

```
         4ISP     ,ITPS    ,K       ,KIN     ,LF      ,MR      ,NCREW   ,    INEDIT
         5NENGS   ,NLISTD  ,NPASS   ,NWL     ,PCHAM   ,Q       ,RHOFU   ,    INEDIT
         6RHOFU2  ,RHOX    ,RHOX2   ,SBODY   ,TOL     ,TOVERC  ,TPRATO  ,    INEDIT
         7TYTAIL  ,VBODY   ,WGROSS  ,NODATA                                   INEDIT
C                                                                             INEDIT
C         NAMELIST TO PRINT OUT INPUT DATA                                    INEDIT
C                                                                             INEDIT
C                                                                             INEDIT
C                                                                             INEDIT
          NAMELIST/DATA2/ALD,FBPAR,IDVELO,ISLB,ISLO,IVACB,IVACO,PERISP,QMXX,  UH
         1 QMXS,SFC,SLVOUT,COPIES,SYNIT,TFCTRB,TFCTRO,TOLMU,TOLTW,           INEDIT
         2 TRATIO,TWLO,TWLOI,WTOUT,FIRE,BOOTW,VCRUSE                         INEDIT
         3 ,NXFOB,PRNTX,FSEC, CLVG, DRNG                                     INEDIT
         4 ,SOLID,AS,BS,SISP,SINERT,SAE,TSBD  ,FLYBCK                        INEDIT
         5 ,WPOREQ,WOREQ,GWREQ                                               INEDIT
         6 ,FBFUEL,CA,CB,WFLYX,RT,R1,R3,SFC1,SFC2,SFC3,ALD1,ALD2,ALD3,VFLY1, INEDIT
         7 VFLY2 ,VFLY3,NODATA                                               INEDIT
         *,PNDX,BUPP,BLOW,STEP,RVAR,PAYX                                     POW
C                                                                             INEDIT
          DIMENSION SAB(392)                                                  INEDIT
          DIMENSION LOCTOS(10),LOCTOE(10), LOCT(10),NTABMX(10)                FIXED
          EQUIVALENCE (LOCTOS,LOCTOS),(LOCTOE,LOCTOE)                         FINI
          DIMENSION LOCTOS(1),LOCTOE(1)                                       FINI
          DATA LOCTOS/  1, 1, 0,  1,2526,3216,5200,7184,9168,11152/           FIXED
         1    ,LOCTOE/102, 1, 0,  1,2615,3277,5261,7245,9229,11211/           FIXED
         2    ,LOCT  /103, 1, 0,  1,2616,3278,5262,7246,9230,    0/           FIXED
         3    ,NTABMX/2000,1000,0,400,111,1954,1954,1954,1954,0   /           FIXED
C                                                                             INEDIT
          DATA FINITY/9988776655./                                            INEDIT
          DIMENSION ALFNU(13)                                                 UG
          LOGICAL NEGO                                                        INEDIT
          NEGO(X)=X.EQ.0. .AND. SIGN(1.,X).LT.0.                              INEDIT
C                                                                             INEDIT
          MFRNT=1                                                             INEDIT
C         ASSUMING 60 WDS IN PC=9                                             INEDIT
          NAA =13075                                                          FIXED
          NAA =13096                                                          FINI
          DO 50 I=1,NAA                                                       INEDIT
     50   AA(I)=-0.                                                           INEDIT
          CALL WRITMS(3,AA,NAA,4)                                             INEDIT
          CALL S(1)                                                           INEDIT
          IN=4                                                                UG
          REWIND IN                                                           UG
          DO 3 I=1,10000                                                      UG
          READ(5,4) ALFNU                                                     UG
          IF(EOF,5) 5,2                                                       UG
        2 WRITE(IN,4) ALFNU                                                   UG
          WRITE(6,6)  ALFNU                                                   UG
        3 CONTINUE                                                            UG
        5 REWIND IN                                                           UG
        4 FORMAT(13A6)                                                        UG
        6 FORMAT(X,13A6)                                                      UG
          OUT=1                                                               INEDIT
C         READ CONTROL CARD INFO                                              INEDIT
          KMAKBD=0                                                            INEDIT
       11 CONTINUE                                                            INEDIT
          MODE= 0                                                             INEDIT
          INTBD=0                                                             INEDIT
          INTRR=0                                                             INEDIT
          INTCS=0                                                             INEDIT
          EXTBD=0                                                             INEDIT
          EXTRR=0                                                             INEDIT
          EXTCS=0                                                             INEDIT
          KBD=0                                                               INEDIT
          KSIZE=0                                                             INEDIT
          ALPHA=6H                                                            INEDIT
          NALPHA=1                                                            INEDIT
          FATAL=0.                                                            INEDIT
     1900 CALL READMS(3,AA,NAA,4)                                             INEDIT
     100  KMOD=0                                                              INEDIT
     102  CONTINUE                                                            INEDIT
          BD=9999                                                             INEDIT
          RR=     9999                                                        INEDIT
```

```
      CS=    9999                                                        INEDIT
      MODCS=9999                                                         INEDIT
      MAKEBD=9999                                                        INEDIT
      SIZING=9999                                                        INEDIT
      READ (IN,XX)                                                       INEDIT
      IF(EOF,IN) 200,300                                                 INEDIT
  200 INTBD=9999                                                         INEDIT
      GO TO 7000                                                         INEDIT
  300 CONTINUE                                                           INEDIT
      IF(KSIZE EQ.0) GO TO 310                                           INEDIT
      DO 301 I=1,392                                                     INEDIT
      SAB(I)=AB(I)                                                       JULY28
  301 AB(I) =AC(I)                                                       JULY28
      READ(IN,DATA3)                                                     INEDIT
      DO 302 I=1,392                                                     INEDIT
  302 AC(I)=AB(I)                                                        INEDIT
      DO 303 I=1,392                                                     INEDIT
  303 AB(I)=SAB(I)                                                       INEDIT
      READ(IN,DATA3)                                                     INEDIT
      READ(IN,DATA2)                                                     INEDIT
      KSIZE=0                                                            INEDIT
  310 CONTINUE                                                           INEDIT
      IF(BD    .NE 9999) GO TO 1100                                      INEDIT
      IF(RR    .NE 9999) GO TO 1200                                      INEDIT
      IF(CS    .NE.9999) GO TO 1300                                      INEDIT
      IF(SIZING NE.9999) GO TO 1310                                      INEDIT
      IF(MODCS  NE.9999) GO TO 1400                                      INEDIT
      IF(MAKEBD.NE.9999) GO TO 1700                                      INEDIT
  321 FORMAT(7H  MODE=,I1)                                               INEDIT
      M=MODE+1                                                           INEDIT
      GO TO (2000,2100,2200,2300,2400,2500,2600,2700,2800),M             INEDIT
C                                                                        INEDIT
C                                                                        INEDIT
C     BD     -EXPECT BASIC DECK OF DATA TO FOLLOW                        INEDIT
 1100 MODE =1                                                            INEDIT
      EXTBD=BD                                                           INEDIT
      EXTRR=0                                                            INEDIT
      EXTCS=0                                                            INEDIT
      IF(SIZING.NE 9999) KSIZE=1                                         INEDIT
      GO TO 1900                                                         INEDIT
C                                                                        INEDIT
C     RR                                                                 INEDIT
 1200 IF(MODE-1) 1220,1250,1210                                          INEDIT
 1210 IF(MODE GE.4 .AND. MODE LE 6) GO TO 1250                           INEDIT
 1220 MODE =4                                                            INEDIT
      EXTRR=RR                                                           INEDIT
      EXTCS=0                                                            INEDIT
      IF(SIZING NE.9999) KSIZE=1                                         INEDIT
      IF(KBD.EQ.0) GO TO 1900                                            INEDIT
C     PRESET AA TO THE CURRENT BASIC DECK ON DISK 1                      INEDIT
      CALL READMS(3,AA,NAA,1)                                            INEDIT
      GO TO 100                                                          INEDIT
 1250 CONTINUE                                                           INEDIT
      WRITE (6,9001)                                                     INEDIT
 9001 FORMAT(100H **DATA SETUP ERROR**  A RR MUST NOT FOLLOW ANY OF THE  INEDIT
     1FOLLOWING CARDS ---  BD/RR/NOMOD/MOD/MODCS   )                     INEDIT
      FATAL=2.                                                           INEDIT
      GO TO 100                                                          INEDIT
C                                                                        INEDIT
C     CS     -                                                           INEDIT
 1300 EXTCS=CS                                                           INEDIT
 1310 IF(SIZING.NE 9999) KSIZE=1                                         INEDIT
      IF(MODCS  NE.9999) GO TO 1400                                      INEDIT
      IF(MAKEBD.NE.9999) GO TO 1700                                      INEDIT
      GO TO 100                                                          INEDIT
C                                                                        INEDIT
C     MOD    -EXPECT 1ST CASE OF A SET OF SERIES-MODIFIED CASES          INEDIT
 1400 KMOD=1                                                             INEDIT
      EXTCS=MODCS                                                        INEDIT
      GO TO 102                                                          INEDIT
```



```
      C                                                                  INEDIT
 1500 CONTINUE                                                           INEDIT
      GO TO 100                                                          INEDIT
 1600 CONTINUE                                                           INEDIT
      GO TO 100                                                          INEDIT
      C                                                                  INEDIT
      C      MAKEBD  MAKES A BD OUT OF SERIES-MODIFIED CASE ON DISK 11   INEDIT
 1700 KMAKBD=1                                                           INEDIT
      KBD=1                                                              INEDIT
      GO TO 100                                                          INEDIT
 1800 CONTINUE                                                           INEDIT
      NTAB=1                                                             INEDIT
      DO 1810 I=1,392                                                    INEDIT
 1810 IF( NOT NEGO(AC(I))) GO TO 1820                                    INEDIT 1820 1830
      GO TO 1830                                                         INEDIT
 1820 IPC=31                                                             INEDIT
      NTABO=392                                                          INEDIT
      WRITE(OUT) IDENT                                                   INEDIT
      WRITE(OUT) ALPHA,(AC(I),I=1,NTABO),DUMMY                           INEDIT
 1830 DO 1840 I=1,392                                                    INEDIT
 1840 IF( NOT NEGO(AB(I))) GO TO 1850                                    INEDIT 1850 1860
      GO TO 1860                                                         INEDIT
 1850 IPC=32                                                             INEDIT
      NTABO=392                                                          INEDIT
      WRITE(OUT) IDENT                                                   INEDIT
      WRITE(OUT) ALPHA,(AB(I),I=1,NTABO),DUMMY                           INEDIT
 1860 DO 1870 I=1,101                                                    POW
 1870 IF(.NOT.NEGO(AD(I))) GO TO 1880                                    INEDIT 1880 1890
      GO TO 1890                                                         INEDIT
 1880 IPC=33                                                             INEDIT
      NTABO =101                                                         POW
      WRITE(OUT) IDENT                                                   INEDIT
      WRITE(OUT) ALPHA,(AD(I),I=1,NTABO),DUMMY                           INEDIT
 1890 CONTINUE                                                           INEDIT
      GO TO 5150                                                         INEDIT 5150
      C                                                                  INEDIT
      C      MODE =0 JUST GOT INDEPENDENT CASE -- ADDED TO CLEAR         INEDIT
 2000 MODE =5                                                            INEDIT
      GO TO 2500                                                         INEDIT
      C                                                                  INEDIT 2500
      C      MODE =1 JUST GOT A BD ADDED TO CLEAR (LEAVE IN AA, PUT ON DISK 8)  INEDIT
 2100 MODE =2                                                            INEDIT
      KBD  =1                                                            INEDIT
      INTBD=INTBD+1                                                      INEDIT
      INTCS=0                                                            INEDIT
      INTRR=0                                                            INEDIT
      CALL WRITMS(3,AA,NAA,1)                                            INEDIT
      WRITE (6,9101) INTBD,EXTBD                                         INEDIT
 9101 FORMAT(6H    BD=I3, 1H(I3,1H))                                     INEDIT
      GO TO 100                                                          INEDIT
      C                                                                  INEDIT
      C      MODE =2 JUST GOT A CS ADDED TO BD                           INEDIT
 2200 IF(KMOD.EQ 1) GO TO 2599                                           INEDIT 2599
      INTCS=INTCS+1                                                      INEDIT
      C      SAVE THIS BD+CS FOR POSSIBLE MODCS COMING UP NEXT           INEDIT
      IF(KMAKBD.EQ 0) GO TO 5000                                         INEDIT 5000
      CALL WRITMS(3,AA,NAA,1)                                            INEDIT
      KMAKBD=0                                                           INEDIT
      C      TRANSPOSE,COMPRESS AND WRITE THIS CASE FOR TABIN            INEDIT
      GO TO 5000                                                         INEDIT 5000
      C      PRESET AA TO THE BASIC DECK IN 1                            INEDIT
 2250 CALL READMS(3,AA,NAA,1)                                            INEDIT
      WRITE (6,9103) INTBD,EXTBD,INTRR,EXTRR,INTCS,EXTCS                 INEDIT
 9103 FORMAT(6H      I3, 1H(I3,1H 5H      I3, 1H(I3,1H) 5H  CS=I3,       INEDIT
     F          1H(I3,1H))                                               INEDIT
      GO TO 100                                                          INEDIT
      C      MODE =3 JUST GOT A CASE ADDED TO RR                         INEDIT
 2300 IF(KMOD EQ 1) GO TO 2599                                           INEDIT 2599
      INTCS=INTCS+1                                                      INEDIT
```

```
      IF(KMAKBD.EQ.0) GO TO 5000                                INEDIT
      CALL WRITMS(3,AA,NAA,1)                                   INEDIT
      KMAKBD=0                                                  INEDIT
      MODE=2                                                    INEDIT
      GO TO 5000                                                INEDIT
 2350 CONTINUE                                                  INEDIT
      WRITE (6,9103) INTBD,EXTBD,INTRR,EXTRR,INTCS,EXTCS        INEDIT
      IF(MODE.EQ.2) GO TO 2250                                  INEDIT
      IF(MODE.EQ.3) CALL READMS(3,AA,NAA,2)                     INEDIT
      IF(MODE.EQ.5) CALL READMS(3,AA,NAA,4)                     INEDIT
      GO TO 100                                                 INEDIT
C                                                               INEDIT
C     MODE =4 JUST GOT A RR ADDED TO CLEAR OR TO BD             INEDIT
 2400 INTRR=INTRR+1                                             INEDIT
      INTCS=0                                                   INEDIT
      MODE =3                                                   INEDIT
      CALL WRITMS(3,AA,NAA,2)                                   INEDIT
      WRITE (6,9102) INTBD,EXTBD,INTRR,EXTRR                    INEDIT
 9102 FORMAT(6H       I3, 1H(I3,1H 5H  RR=I3, 1H(I3,1H))        INEDIT
      GO TO 100                                                 INEDIT
C                                                               INEDIT
C     MODE =5 JUST GOT AN INDEPENDENT CASE ADDED TO CLEAR       INEDIT
 2500 GO TO 2300                                                INEDIT
 2550 GO TO 2300                                                INEDIT
C                                                               INEDIT
C     MODE =6 JUST GOT A CASE ADDED TO CS                       INEDIT
 2599 MODE=6                                                    INEDIT
 2600 INTCS=INTCS+1                                             INEDIT
      CALL WRITMS(3,AA,NAA,3)                                   INEDIT
      GO TO 5000                                                INEDIT
 2650 CALL READMS(3,AA,NAA,3)                                   INEDIT
      WRITE (6,9103) INTBD,EXTBD,INTRR,EXTRR,INTCS,EXTCS        INEDIT
      GO TO 100                                                 INEDIT
C                                                               INEDIT
 2700 CONTINUE                                                  INEDIT
 2800 CONTINUE                                                  INEDIT
      GO TO 100                                                 INEDIT
 5000 CONTINUE                                                  INEDIT
C     SET UP TABLES FOR EACH PART CASE                          INEDIT
      MPRINT=MPRNT                                              INEDIT
      IPC=0                                                     INEDIT
      GO TO 5200                                                INEDIT
 5100 IF(IPC.LT.30) GO TO 5200                                  INEDIT
      GO TO 1800                                                INEDIT
 5150 CONTINUE                                                  INEDIT
      GO TO (100,2250,2350,100,2350,2650),MODE                  INEDIT
 5200 IPC=IPC+1                                                 INEDIT
      NTOT=1                                                    INEDIT
      IF(IPC-10) 5600,5600,5400                                 FIXED
 5300 DO 5350 I=1,60                                            INEDIT
 5350 IF(.NOT.NEGO(MBO(I))) GO TO 5560                          INEDIT
      GO TO 5100                                                INEDIT
 5400 L=IPC-10                                                  INEDIT
      LL=L+980                                                  INEDIT
      LLL=1                                                     INEDIT
      LLLL=0                                                    INEDIT
      DO 5500 I=L,LL,20                                         INEDIT
      AA(LLL)=SREF(I)                                           INEDIT
      IF(.NOT.NEGO(AA(LLL))) LLLL=1                             INEDIT
 5500 LLL=LLL+1                                                 INEDIT
      IF(LLLL.EQ.0) GO TO 5100                                  INEDIT
 5550 JTO=1                                                     INEDIT
      KTO=50                                                    INEDIT
      NTABO=50                                                  INEDIT
 5560 NT=0                                                      INEDIT
      NTAB=1                                                    INEDIT
      GO TO 7000                                                INEDIT
```

```
      5600 IF(IPC.EQ.3) GO TO 5100                                    INEDIT
           JTO=  LOCTOS(IPC)                                          INEDIT
           KTO=  LOCTOE(IPC)                                          INEDIT
           NTABO=KTO-JTO+1                                            INEDIT
           NTOT =0                                                    INEDIT
           GO TO (5701,6000,5100,6800,5705,5706,5707,5708,5709,5300),IPC FIXED
      5701 CALL PACKER(TCLA  ,2100,70)                                INEDIT
           GO TO 5800                                                 INEDIT
      5705 CALL PACKER(TCONA ,600 ,30)                                INEDIT
           GO TO 5800                                                 INEDIT
      5706 CALL PACKER(TCLCD1,1922,62)                                INEDIT
           GO TO 5800                                                 INEDIT
      5707 CALL PACKER(TCLCD2,1922,62)                                INEDIT
           GO TO 5800                                                 INEDIT
      5708 CALL PACKER(TCLCD3,1922,62)                                INEDIT
           GO TO 5800                                                 FIXED
      5709 CALL PACKER( THSF,1922,62)                                 FIXED
      5800 IF(NT GT.0) GO TO 5820                                     INEDIT
           DO 5810 I=JTO,KTO                                          INEDIT
      5810 IF(.NOT NEGO(AA(I))) GO TO 5820                            INEDIT
           GO TO 5100                                                 INEDIT
      5820 JT=LOCT(IPC)                                               INEDIT
           GO TO 5900                                                 INEDIT
      5850 JT=1                                                       INEDIT
           IF(NT.EQ.0) GO TO 5100                                     INEDIT
      5900 KT=JT+NT-1                                                 INEDIT
           IF(NT EQ.0) NTOT=1                                         INEDIT
           NTAB=NT+NTOT                                               INEDIT
           IF(NTAB.LE.NTABMX(IPC)) GO TO 7000                         INEDIT
           WRITE (6,5901) NTAB,IPC,NTABMX(IPC)                        INEDIT
      5901 FORMAT(36H ***FATAL INPUT ERROR***  TOO MUCH (,I4),26H) DATA FOR T INEDIT
          FHIS PARTCACE (,I2,17H) - ALLOWED ONLY ,I4)                 INEDIT
           FATAL=1.                                                   INEDIT
           GO TO 5100                                                 INEDIT
     C                                                                INEDIT
      6000 NT1=-3                                                     INEDIT
           DO 6100 I=1,39,2                                           INEDIT
           IF(NEGO(T1(I)))  GO TO 6100                                INEDIT
           NT1=NT1+4                                                  INEDIT
           AA(NT1)  =(I+1)/2                                          INEDIT
           AA(NT1+1)=1.                                               INEDIT
           AA(NT1+2)=T1(I)                                            INEDIT
           AA(NT1+3)=T1(I+1)                                          INEDIT
      6100 CONTINUE                                                   INEDIT
           DO 6120 I=1,39,2                                           INEDIT
           IF(NEGO(V1(I)))  GO TO 6120                                INEDIT
           NT1=NT1+4                                                  INEDIT
           AA(NT1)  =(I+1)/2                                          INEDIT
           AA(NT1+1)=2.                                               INEDIT
           AA(NT1+2)=V1(I)                                            INEDIT
           AA(NT1+3)=V1(I+1)                                          INEDIT
      6120 CONTINUE                                                   INEDIT
           DO 6140 I=1,39,2                                           INEDIT
           IF(NEGO(W1(I)))  GO TO 6140                                INEDIT
           NT1=NT1+4                                                  INEDIT
           AA(NT1)  =(I+1)/2                                          INEDIT
           AA(NT1+1)=5.                                               INEDIT
           AA(NT1+2)=W1(I)                                            INEDIT
           AA(NT1+3)=W1(I+1)                                          INEDIT
      6140 CONTINUE                                                   INEDIT
           IF(NEGO(GAM1 ))  GO TO 6220                                INEDIT
           NT1=NT1+4                                                  INEDIT
           AA(NT1)  =1.                                               INEDIT
           AA(NT1+1)=3.                                               INEDIT
           AA(NT1+2)= GAM1(1)                                         INEDIT
           AA(NT1+3)= GAM1(2)                                         INEDIT
      6220 IF(NEGO(ALT1 ))  GO TO 6230                                INEDIT
```

72 G.01-46

```
      NT1=NT1+4                                           INEDIT
      AA(NT1)  =1.                                        INEDIT
      AA(NT1+1)=4.                                        INEDIT
      AA(NT1+2)= ALT1(1)                                  INEDIT
      AA(NT1+3)= ALT1(2)                                  INEDIT
6230  IF(NEGO(PSI1  ))  GO TO 6240                        INEDIT  6240
      NT1=NT1+4                                           INEDIT
      AA(NT1)  =1.                                        INEDIT
      AA(NT1+1)=6                                         INEDIT
      AA(NT1+2)= PSI1(1)                                  INEDIT
      AA(NT1+3)= PSI1(2)                                  INEDIT
6240  IF(NEGO(RHO1  ))  GO TO 6250                        INEDIT  6250
      NT1=NT1+4                                           INEDIT
      AA(NT1)=1.                                          INEDIT
      AA(NT1+1)=7.                                        INEDIT
      AA(NT1+2)= RHO1(1)                                  INEDIT
      AA(NT1+3)= RHO1(2)                                  INEDIT
6250  IF(NEGO(MU1   ))  GO TO 6260                        INEDIT  6260
      NT1=NT1+4                                           INEDIT
      AA(NT1)=1.                                          INEDIT
      AA(NT1+1)=8.                                        INEDIT
      AA(NT1+2)= MU1(1)                                   INEDIT
      AA(NT1+3)= MU1(2)                                   INEDIT
6260  IF(NEGO(H1    ))  GO TO 6270                        INEDIT  6270
      NT1=NT1+4                                           INEDIT
      AA(NT1)=1.                                          INEDIT
      AA(NT1+1)=10.                                       INEDIT
      AA(NT1+2)= H1(1)                                    INEDIT
      AA(NT1+3)= H1(2)                                    INEDIT
6270  IF(NEGO(SQ1   ))  GO TO 6280                        INEDIT  6280
      NT1=NT1+4                                           INEDIT
      AA(NT1)=1.                                          INEDIT
      AA(NT1+1)=11.                                       INEDIT
      AA(NT1+2)= SQ1(1)                                   INEDIT
      AA(NT1+3)= SQ1(2)                                   INEDIT
6280  IF(NEGO(TIM1  ))  GO TO 6290                        INEDIT  6290
      NT1=NT1+4                                           INEDIT
      AA(NT1)=1.                                          INEDIT
      AA(NT1+1)=9.                                        INEDIT
      AA(NT1+2)= TIM1(1)                                  INEDIT
      AA(NT1+3)= TIM1(2)                                  INEDIT
6290  CONTINUE                                            INEDIT
      NT1=NT1+3                                           INEDIT
      NT2=NT1-3                                           INEDIT
      IF(NEGO(PAYOFF)) GO TO 6510                         INEDIT  6510
      NT2=NT2+4                                           INEDIT
      AA(NT2)   = PAYOFF                                  INEDIT
      AA(NT2+1)= ABS(PAYOFF(2))                           INEDIT
      AA(NT2+2)= SIGN(2.,PAYOFF(2))                       INEDIT
      AA(NT2+3)= PAYOFF(3)                                INEDIT
6510  DO 6550 I=1,22,3                                    INEDIT
      IF(NEGO(CN1(I))) GO TO 6550                         INEDIT  6550
      NT2=NT2+4                                           INEDIT
      AA(NT2)   = CN1(I)                                  INEDIT
      AA(NT2+1)= CN1(I+1)                                 INEDIT
      AA(NT2+2)=0.                                        INEDIT
      AA(NT2+3)= CN1(I+2)                                 INEDIT
6550  CONTINUE                                            INEDIT
      DO 6580 I=1,39,2                                    INEDIT
      IF(NEGO(CT1(I))) GO TO 6580                         INEDIT  6580
      NT2=NT2+4                                           INEDIT
      AA(NT2)   =(I+1)/2                                  INEDIT
      AA(NT2+1)=ABS(CT1(I))                               INEDIT
      AA(NT2+2)=SIGN(1.,CT1(I))                           INEDIT
      AA(NT2+3)=CT1(I+1)                                  INEDIT
6580  CONTINUE                                            INEDIT
      NT2=NT2-NT1+3                                       INEDIT
      NT=NT1+NT2                                          INEDIT
      IF(NT1 EQ.0) GO TO 6610                             INEDIT  6610
      JTOT(1)=1                                           INEDIT
```

119

```
      IF(NT2 EQ 0) GO TO 6600                                           INEDIT
      JTOT(2)=NT1+1                                                     INEDIT
      NTOT=3                                                            INEDIT
      GO TO 5850                                                        INEDIT
 6600 NTOT=2                                                            INEDIT
      GO TO 5850                                                        INEDIT
 6610 IF(NT2 EQ.0) GO TO 6620                                           INEDIT
      JTOT(1)=0                                                         INEDIT
      JTOT(2)=1                                                         INEDIT
      NTOT=3                                                            INEDIT
      GO TO 5850                                                        INEDIT
 6620 NTOT=1                                                            INEDIT
      GO TO 5850                                                        INEDIT
C                                                                       INEDIT
 6800 NT=-3                                                             INEDIT
      XN=0.                                                             INEDIT
      DO 6850 I=1,115,6                                                 INEDIT
      XN=XN+1000000.                                                    INEDIT
      IF(NEGO(PHI(I))) GO TO 6850                                       INEDIT
      NT=NT+4                                                           INEDIT
      AA(NT)  = XN+PHI(I)*1000.+PHI(I+1)                                INEDIT
      AA(NT+1)= PHI(I+2) *1000000. + PHI(I+3)                           INEDIT
      AA(NT+2)= PHI(I+4)                                                INEDIT
      AA(NT+3)= PHI(I+5)                                                INEDIT
 6850 CONTINUE                                                          INEDIT
      NT=NT+3                                                           INEDIT
      JTOT(1)=1                                                         INEDIT
      NTOT   =2                                                         INEDIT
      GO TO 5850                                                        INEDIT
C                                                                       INEDIT
 7000 CONTINUE                                                          INEDIT
C     IF(FATAL.NE 0 ) GO TO 7300                                        INEDIT
      WRITE (OUT) IDENT                                                 INEDIT
      IF(INTBD EQ.9999) GO TO 7200                                      INEDIT
      IF(NT.EQ 0) GO TO 7100                                            INEDIT
      WRITE (OUT) ALPHA,(AA(I),I=JTO,KTO),(ITOT(I),I=1,NTOT),           INEDIT
     W                     (AA(I),I=JT ,KT )                            INEDIT
      GO TO 5100                                                        INEDIT
 7100 WRITE (OUT) ALPHA,(AA(I),I=JTO,KTO),DUMMY                         INEDIT
      GO TO 5100                                                        INEDIT
 7200 IF(FATAL.NE.0.) GO TO 7400                                        INEDIT
      REWIND 1                                                          INEDIT
      CALL S(2)                                                         INEDIT
      CALL S(8)                                                         INEDIT
      RETURN                                                            INEDIT
 7300 IF(INTBD NE 9999) GO TO 5100                                      INEDIT
 7400 CONTINUE                                                          INEDIT
      STOP                                                              INEDIT
      END                                                               INEDIT
```

72 G 01-46

Subroutine PACKER

Purpose

Subroutine PACKER packs univariant and bivariate tables for inclusion on file 1.

Description

PACKER is called by INEDIT to process the input data tables into a form that is stored on file 1.

```
      SUBROUTINE PACKER(A,LNA,MWT)                                          INEDIT
      DIMENSION A(1)                                                        INEDIT
      COMMON/TT/INTBD,INTRR,INTCS,IPC,NALPHA,NTABO,NTAB,NT,NTOT,JTOT(31)    INEDIT
      NT=1                                                                  INEDIT
      LL=LNA-MWT+1                                                          INEDIT
      MWTM1=MWT-1                                                           INEDIT
      NOT=0                                                                 INEDIT
      DO 1000 I=1,LL,MWT                                                    INEDIT
      NOT=NOT+1                                                             INEDIT
      IF(A(I).EQ.0. .AND. SIGN(1.,A(I)).LT.0.) GO TO 600                    INEDIT  600
      JTOT(NOT)=NT                                                          INEDIT
300   K=I+MWT-2                                                             INEDIT
      DO 400 J=I,K,2                                                        INEDIT
      IF(A(J).EQ.0. .AND. SIGN(1.,A(J)).LT.0.) GO TO 500                    INEDIT  500
      A(NT)=  A(J)                                                          INEDIT
      A(NT+1)=A(J+1)                                                        INEDIT
400   NT=NT+2                                                               INEDIT
500   NTOT=NOT                                                              INEDIT
      GO TO 1000                                                            INEDIT  1000
600   JTOT(NOT)=0                                                           INEDIT
1000  CONTINUE                                                              INEDIT
      NT=NT-1                                                               INEDIT
      NTOT=NTOT+1                                                           INEDIT
      RETURN                                                                INEDIT
      END                                                                   INEDIT
```

20 OCT 72 6 01-46

SUBRØUTINE
S

123

```
      SUBROUTINE S(L)                                                      INEDIT
      COMMON/TT/II(3)                                                      INEDIT
      CALL SECOND(T)                                                       INEDIT
      DT=T-TO                                                              INEDIT
      WRITE(6,1) L,T,DT                                                    INEDIT
    1 FORMAT(25H ***MARK TIME AT LOCATION,I2,3H T=,F6.3,6H   DT=F6 3)      INEDIT
      TO=T                                                                 INEDIT
      RETURN                                                               INEDIT
      END                                                                  INEDIT
```

20 OCT 72 G.01-46

124

Input Scan (GEINP) 4

CONTENTS

SUBRØUTINE GEINP

GEINP

Purpose

GEINP reads various data sets from the input tape file, puts some in common blocks, and calls other routines that process data before storing.

Description

GEINP is an overlay program on CDC equipment whereas it is a subroutine on the UNIVAC system.

The data that GEINP uses is fetched by calling TABIN according to "Part Case" number. (TABIN reads the data from unit 1. For a description of how data gets on unit 1 see Subroutine INEDIT). The part cases that are read in this routine (in the same order as read) are:

Part Case Number	Description	Common Blocks
1	GLOBAL, data and univariant tables (subroutine SPLICØ spline fits tables and stores them in common block/TABLE/).	/GLOBAL/ /TBLE/
11 → 30	BLOCKS OF ARC DATA read in and processed by FXDAT and then stored on random file 9, records 1 through 20.	/ARCDAT/
6, 7, 8	Bivariate aerodynamics tables read in and processed by subroutine BLICØ and stored on random files 33, 34, and 35, respectively.	/BICUBE/
9	Bivariate Airbreather thrust table, read in and processed by subroutine BLICØ and stored on random file 36.	/BICUBE/
10	SIZING Data is read in and stored in /SIZING/	/SIZING/
2	Boundary condition data is read in and processed in BNDRYC. It is stored after processing on random file 9, record 21.	

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ARCDA	S_{ref}	M	Aerodynamic reference area (FT^2)	/ARCDAT/(	1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SDINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF
BNARR		W	Packed array of boundary conditions	/GEINP /(*	)	GEINP	W	BNARR
DTNC	$\Delta\tau$	M	Integration interval (SEC)	/ARCDAT/(	5)	BNTG	I	DTNC
						FNTG	I	DTNC
						GEINP	M	DTNC
						PROPIN	I	DTNC
DTPI		M	Print frequency for trajectory	/ARCDAT/(	6)	FNTG	I	DTPI
						GEINP	M	DTPI
ER	E_R	I	Earth radius (FT)	/GLOBAL/(	2)	COORDS	I	ER
						CRASH	I	REM
						EQUA3	I	ER
						GEINP	I	ER
						PADS1	I	ER
						PDBC	I	ER
						SOMG	I	ER
						TRTOSZ	I	ER
G	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ID		I	A four word array containing the basic deck, reference run, case and part case numbers in that order	/GLOBAL/(	21)	BLICO	I	ID
						FRENCH	I	ID
						GEINP	I	ID
						PADS1	O	ID
						PRINT	I	ID
						SDINP	I	ID
						TOPM	I	ID
						VEHDF	I	ID
IFATAL		O	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADS1	I	IFATAL
						SDINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
IG	g_r	O	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FN3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
ITAB		I	A 20 word array containing the number of nonzero state initial conditions specified at the beginning of each subarc.	/GLOBAL/(	45)	GEINP	I	ITAB
						SDINP	I	ITAB
						SIZIN	I	ITAB
JAER		I	Aerodynamic model option flag	/ARCDAT/(	9)	AEROCO	I	JAER
						EQUA3	I	JAER
						GEINP	I	JAER
						OUT	I	JAER
						PROPB	I	JAER
						PROPIN	I	JAER
						VT	I	JAER
JAF		W	Saved aero option codes for each arc	/GEINP /(*	)	GEINP	W	JAF
JPRO		I	Propulsion model option flag	/ARCDAT/(	10)	EQUA3	I	JPRO
						GEINP	I	JPRO
						IMPUL	I	JPRO
						MODELA	I	JPRO
						PROPB	I	JPRO
						PROPIN	I	JPRO
KTAB		I	A 20 word array containing the number of state target conditions specified at the end of each subarc.	/GLOBAL/(	25)	GEINP	I	KTAB
						SDINP	I	KTAB
						SIZIN	I	KTAB
LUM		I	Program control flag. LUM = 0: Steepest descent only; LUM = 1 Steepest descent and adjoint transformation stored on tape; LUM = 2. Steepest descent and QL; LUM = 3: QL only.	/GLOBAL/(	6)	AST3	I	LUM
						FNTG	I	LUM
						GEINP	I	LUM
						PADS1	M	LUM
						SDINP	I	LUM
						TOPM	M	LUM
MAEA		I	Curve number	/ARCDAT/(	18)	EQUA3	I	MAEA
						GEINP	I	MAEA
						PROPB	I	MAEA
						PROPIN	I	MAEA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
NARC	N_3	M	Number of subarcs in the problem.	/GLOBAL/(	18)	FNTG	I	NARC
						GEINP	M	NARC
						PROPIN	I	NARC
						SDINP	I	NARC
						SIZIN	I	NARC
OMGZ	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
						CRASH	I	OMEGA
						DER3A	I	OMGZ
						EQUA3	I	OMGZ
						GEINP	I	OMGZ
						MODELA	I	OMGZ
						MODELB	I	OMGZ
						PDBC	I	OMGZ
						PDY3A	I	OMGZ
						SDINP	I	OMGZ
						TOPM	I	OMGZ
PSIRF	ψ_r	I	Reference azimuth. (DEG)	/GLOBAL/(	68)	GEINP	I	PSIRF
						REU3	I	PSIRF
						SDINP	M	PSIRF
TARG		I	Packed array of targets	/GEINP /(*	)	GEINP	I	TARG
XLAMRF	ρ_r	I	Reference latitude. (DEG)	/GLOBAL/(	4)	CRASH	M	RHOO
						GEINP	I	XLAMRF
						REU3	I	XLAMRF
						SDINP	M	XLAMRF
YMURF	μ_r	I	Reference longitude. (DEG)	/GLOBAL/(	5)	CRASH	M	UMUO
						GEINP	I	YMURF
						REU3	I	YMURF
						SDINP	M	YMURF
.UN06.		O	File of all output data	/.UN06./($	)	BLICO	O	.UN06.
						BNORYC	O	.UN06.
						CRASH	O	.UN06.
						FRENCH	O	.UN06.
						FXDAT	O	.UN06.
						GEINP	O	.UN06.
						HUNT	O	.UN06.
						INEDIT	O	.UN06.
						ITER8	O	.UN06.
						MODELA	O	.UN06.
						MOMJ	O	.UN06.
						MPSI	O	.UN06.
						OUT	O	.UN06.
						PAYO2	O	.UN06.
						PRINT	O	.UN06.
						PRINTV	O	.UN06.
						PRINTW	O	.UN06.
						PRITEQ	O	.UN06.
						PRITVA	O	.UN06.
						PROPIN	O	.UN06.
						PROTHR	O	UN06
						PRWTSM	O	UN06
						RANGE	O	.UN06.
						S	O	.UN06.
						SDINP	O	.UN06.
						SIZE	O	UN06
						SIZIN	O	UN06
						SIZOUT	O	.UN06.
						SOLVE	O	.UN06.
						SPLICO	O	UN06.
						SPLIZ	O	.UN06.
						SPLYNE	O	UN06.
						SSSP	O	UN06.
						STAU	O	UN06
						STPIT	O	.UN06.
						SUMOUT	O	.UN06.
						TABIN	O	.UN06
						TEST	O	.UN06.
						VEHDF	O	.UN06
						WTSCH	O	.UN06.
						WTVOL	O	.UN06.

```
      PROGRAM GEINP                                                   GEINP
C         INPUT  ROUTINE                                              COMN
C        READS IN   GLOBAL, ARCDAT, TABLE, BIVARIATE DATA,SIZING(PHASE I COMN
C        AND BOUNDARY CONDITION  DATA  AND  PROCESSES IT              COMN
C                                                                     COMN
      COMMON/GLOBAL/                                                  GLOBAL
     *GR         ,ER         ,OMGZ      ,XLAMRF     ,YMURF     ,LUM   GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC        ,NBRAN     ,NFARC     ,ID(4) GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                            GLOBAL
     *,GM          ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20) GLOBAL
     *,ITPSO ,KSOL      ,KGLOBL(8)                                    RETAP
      COMMON/ARCDAT/                                                  ARCDAT
     *SREF       ,EJ         ,XISP       ,TMULT      ,DTNC       ,DTPI      , ARCDAT
     *IATM       ,IMODE      ,JAER       ,JPRO       ,QMAX       ,GMAX      , ARCDAT
     *XLMAX      ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX      ,MAEA      , ARCDAT
     *MAEB       ,MAEC       ,MAED       ,MAEE       ,MAEF       ,MAEG      , ARCDAT
     *MT         ,MISP       ,MXCG       ,MZCG       ,MWDA       ,MWDB      , ARCDAT
     *MDB        ,XCGR       ,ZCGR       ,XE         ,ZE         ,XT        , ARCDAT
     *DREF       ,MCND       ,RHOB       ,QMULT      ,REMAX               ARCDAT
     * ,FRATE   , ARCD(9)                                             RETAP
      DIMENSION  ARCDA(40)                                            ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                         ARCDAT
      DIMENSION  BNARR(400), TARG(100)                                GEINP
      EQUIVALENCE (BNARR(301),TARG(1)),(A,III)                        GEINP
      DIMENSION G(1),TEMP(51),A(1000),III(100),IG(1)                  P014
      EQUIVALENCE(GR,G) ,(GR,IG)                                      GEINP
      COMMON/LASTAB/ LSTWD                                            GEINP
      COMMON/TABLE/  TABLE(2003)                                      GEINP
      COMMON/TBLE/TBLE(2000)                                          GEINP
      REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                         SIZING
      COMMON /SIZING/                                                 SIZING
C         PHASE II SIZING PARAMERERS                                  SIZING
     *TZ,     VV(3),     OP(14),   EROR,    PZ(5),     VO,      SW(20), SIZING
     *SV(28),  SQ(37,5), SE(11),   TLAT,    TLNG,                     SIZING
C        PHASE I SIZING PARAMERERS                                    SIZING
     *WBO,     WLDO,     DWEB,     DWEO,     TOLWT,    WPB,     TWRAT2, SIZING
     *BK1,     BK2,      BK3,      BK4,      ISIZE,    TRAFLG,  TWRATO, SIZING
     *OK1,     OK2,      OK3,      OK4,      PRFLG,    IPASS,   IPSMAX, SIZING
     *AEXIT,   TVACO,    NO,       WFO,      IDVEL,    ISPO,    ISPB,   SIZING
     *XPL,     TVACB,    NNB,      WEO,      WEB,      WD,      WLO,    SIZING
     *DVO,     DVB,      MUB,      MUO,      VSTG,     WPO              SIZING
     *,JTYP,   BECO ,    BSTG ,    ORBI,    ITNBW , ITNOW  ,          SIZING
     * SVDPSQ  , SVDCON  ,IHUNT   ,IOPSTG ,ISZO(19)                    UH
      EQUIVALENCE(SIZ,WBO    )                                        PH1SZ
      DIMENSION SIZ(1)                                                PH1SZ
      DIMENSION JAF(80)                                               GEINP
      DIMENSION IFACT(4)                                              GEINP
      DATA IFACT/10HST.DESCENT ,10HST D+TAPE  ,10HST.D.+QLIN ,10HQUAS.LI GEINP
     1N.   /                                                          GEINP
      LOGICAL IFAT                                                    GEINP
      IERR = 0                                                        GEINP
      MN= 1                                                           GEINP
C      I  FETCH  GLOBAL  AND  UNIVARIATE TABLE  DATA                  COMN
      CALL TABIN(A,1,G,102,TBLE,1400,ID,MN,1,IEOD)                    NOS
      IFATAL=.FALSE.                                                  P014
C      I-A   FIX  FLOATING PT. NOS.                                   COMN
      IG(6)=G(6)                                                      GEINP
      IG(9)=G(9)                                                      GEINP
      IG(10)=G(10)                                                    GEINP
      IG(11)=G(11)                                                    GEINP
      IG(12)=0                                                        P014
      IG(13)=0                                                        P014
      IG(14)=0                                                        P014
      IG(15)=0                                                        P014
      IG(18)= G(18)                                                   GEINP
      IG(69)=G(69)                                                    GEINP
      IG(70)=G(70)                                                    GEINP
      IG(71)=G(71)                                                    GEINP
      IG(72)=G(72)                                                    GEINP
      IG(93)=G(93)                                                    NOS
      IF(IEOD)10,20,200                                               GEINP
   10 IERR=1                                                          GEINP
      WRITE(6,210) MN                                                 GEINP
      GO TO 50                                                        GEINP
```

20 OCT 72 G.01-46

```
   20 IF(G(1)*G(2) NE.0) GO TO  30                                   GEINP
      WRITE(6,220) G(1),G(2)                                          GEINP
      IFATAL=.TRUE.                                                   GEINP
   30 CONTINUE                                                        GEINP
      IF(NARC.LT.1) NARC=1                                            NOS
      IF(NARC.GT.20) NARC=20                                          NOS
      WRITE(6,40) IFACT(LUN+1),GR,ER,OMGZ,XLAMRF,YMURF,PSIRF          GEINP
   40 FORMAT(1H1,36X, 45HMCDONNELL DOUGLAS ASTRONAUTICS COMPANY - WEST/ GEINP
     145X, 29HTABTOP COMPUTER PROGRAM-P1551//5X,11HGLOBAL DATA/6X,   GEINP
     215HTHIS WILL BE A A10, 4H RUN//6X, 8HGRAVITY=F9.5,14H EARTH RADIUS GEINP
     3=                                                               GEINP
     4F12 2,14H ROTAT   RATE=E15.6,13H REF. LATIT =F9.4,14H REF. LONG. GEINP
     5=F9.4/6X, 9HREF.AZIM=F9.4)                                      GEINP
C                                                                     COMN
C       I-C    SPLINE  FIT  UNIVARIATE  TABLES                        COMN
      CALL  SPLICO                                                    GEINP
C                                                                     COMN
C       II     READ  IN  ARC DATA ,PROCESS  AND STORE ON RANDOM FILE  COMN
   50 MN=11                                                           GEINP
      WRITE(6,60)                                                     GEINP
   60 FORMAT(1H1)                                                     GEINP
      CALL TABIN(DUMMY,1,ARCDA,51,RUMMY,1,ID,MN,0,IEOD)               FRAT
      IF(IEOD .EQ. 0) GO TO  70                                       GEINP
      IERR = 1                                                        GEINP
      WRITE(6,210) MN                                                 GEINP
      GO TO 140                                                       GEINP
   70 DO 80 I=1,51                                                    FRAT
   80 TEMP(I) = ARCDA(I)                                              GEINP
      CALL FXDAT(1)                                                   GEINP
      CALL WRITMS(9,ARCDA,51,1)                                       NOS
C       II-A    SAVE   KEY  OPTION FLAGS  AND SET  NUMBERS            COMN
      JAF(1)=JAER                                                     GEINP
      JAF(2)=MAEA                                                     GEINP
      JAF(3)= JPRO                                                    FIXED
      JAF(4)= MWDB                                                    FIXED
      IF(NARC.EQ.1) GO TO 140                                         GEINP
      DO 130 IARC=2,NARC                                              GEINP
      CALL TABIN(DUMMY,1,ARCDA,51,RUMMY,1,ID,IARC+10,0,IEOD)          FRAT
      IF(IEOD .EQ. 0) GO TO 90                                        GEINP
      CALL  STPIT(98)                                                 GEINP
   90 CONTINUE                                                        GEINP
      DO  100  I=1,51                                                 FRAT
      IF(ABS(ARCDA(I)) .NE. 0.) GO TO 100                             GEINP
      IF(SIGN(1., ARCDA(I)) .LT. 0.) ARCDA(I)= TEMP(I)                GEINP
  100 CONTINUE                                                        GEINP
      IF(DTPI.EQ.0.) DTPI=1.                                          PO14
      IF(DTNC.EQ.0.) DTNC=1.                                          NOS
  110 DO  120   I=1,51                                                FRAT
  120 TEMP(I) = ARCDA(I)                                              GEINP
      CALL FXDAT(IARC)                                                GEINP
      CALL WRITMS(9,ARCDA,51,IARC)                                    FRAT
      KO=(IARC-1)*4                                                   GEINP
      JAF(KO+1)=JAER                                                  GEINP
      JAF(KO+2)=MAEA                                                  GEINP
      JAF(KO+3)= JPRO                                                 FIXED
      JAF(KO+4)= MWDB                                                 FIXED
  130 CONTINUE                                                        GEINP
  140 CONTINUE                                                        GEINP
C       III     TEST FOR  BIVARIATE TABLES  AND BICUBIC SPLINE FIT    COMN
      DO 155 I=1,NARC                                                 FIXED
      KO=4*(I-1)                                                      GEINP
      IF(JAF(KO+1).NE.2) GO TO 150                                    GEINP
      CALL BLICO(JAF(KO+2))                                           GEINP
  150 CONTINUE                                                        GEINP
      IF(JAF(KO+3).NE.2) GO TO 155                                    FIXED
      CALL BLICO( JAF(KO+4))                                          FIXED
  155 CONTINUE                                                        FIXED
C       IV     READ  IN  SIZING DATA  AND  SET FLAGS                  COMN
      MN=10                                                           FIXED
      CALL TABIN(DUMMY,1,SIZ,70,RUMMY,1,ID,MN,0,IEOD)                 PHISZ
      IF(IEOD.NE.0) GO TO 152                                         PHISZ
```

```
      CALL IPR(3HSIZ  ,SIZ,1,70,0)                                PHISZ
      IPASS=1                                                     PHISZ
      JTYP=SIZ(42)                                                UH
      IF(SIZ(21) EQ.0.) SIZ(21)=6.                                FINI
      IHUNT= SIZ(50)                                              POW
      IOPSTG = SIZ(51)                                            OS
      GO TO 153                                                   PHISZ
  152 JTYP =0                                                     PHISZ
  153 CONTINUE                                                    PHISZ
      MN=2                                                        GEINP
C       V                                                         COMN
C READ IN  BOUNDARY CONDITION DATA                                GEINP
      CALL TABIN(DUMMY,1,RUMMY,1,A,1000,  ID,MN,0,IEOD)           GEINP
      IF(IEOD .NE. 0) GO TO 160                                   GEINP
      IF(III EQ.3) GO  TO  170                                    GEINP
  160 WRITE(6,210) MN                                             GEINP
      GO TO 190                                                   GEINP
  170 ILC1 = 4                                                    GEINP
      ILC2 = 3 + III(3)                                           GEINP
      DO 180 I=1,400                                              GEINP
  180 BNARR(I)=0.                                                 GEINP
      IFAT= .FALSE.                                               GEINP
      CALL BNDRYC(A,ILC1,ITAB,BNARR,300,ILC2-1,IFAT   )           GEINP
      IF(IFAT) IERR=1                                             GEINP
      IFAT =.FALSE.                                               GEINP
      CALL TARGCN(A,ILC2,KTAB,TARG,100,LSTWD,IFAT)                GEINP
      IF(IFAT)GO TO 190                                           GEINP
      IF(IERR.EQ.1) GO TO 190                                     GEINP
      CALL IPR(4HKTAB,A,KTAB,NARC+1,1)                            GEINP
      CALL IPR(4HITAB,A,ITAB,NARC+1,1)                            GEINP
      CALL IPR(5HBNARR,BNARR,I,100,0)                             GEINP
      CALL IPR(4HTARG,TARG,I,50,0)                                GEINP
      CALL WRITMS(9,BNARR,400,21)                                 GEINP
      RETURN                                                      GEINP
  190 CALL  STPIT(99)                                             GEINP
  200 CALL EXIT                                                   GEINP
  210 FORMAT(26H  INPUT ERROR IN PART-CASEI3)                     GEINP
  220 FORMAT(25H  GLOBAL DATA MISSING GR=E17.8,4H ER=E17.8)       GEINP
      END                                                         GEINP
```

20 OCT 72 6 01-46

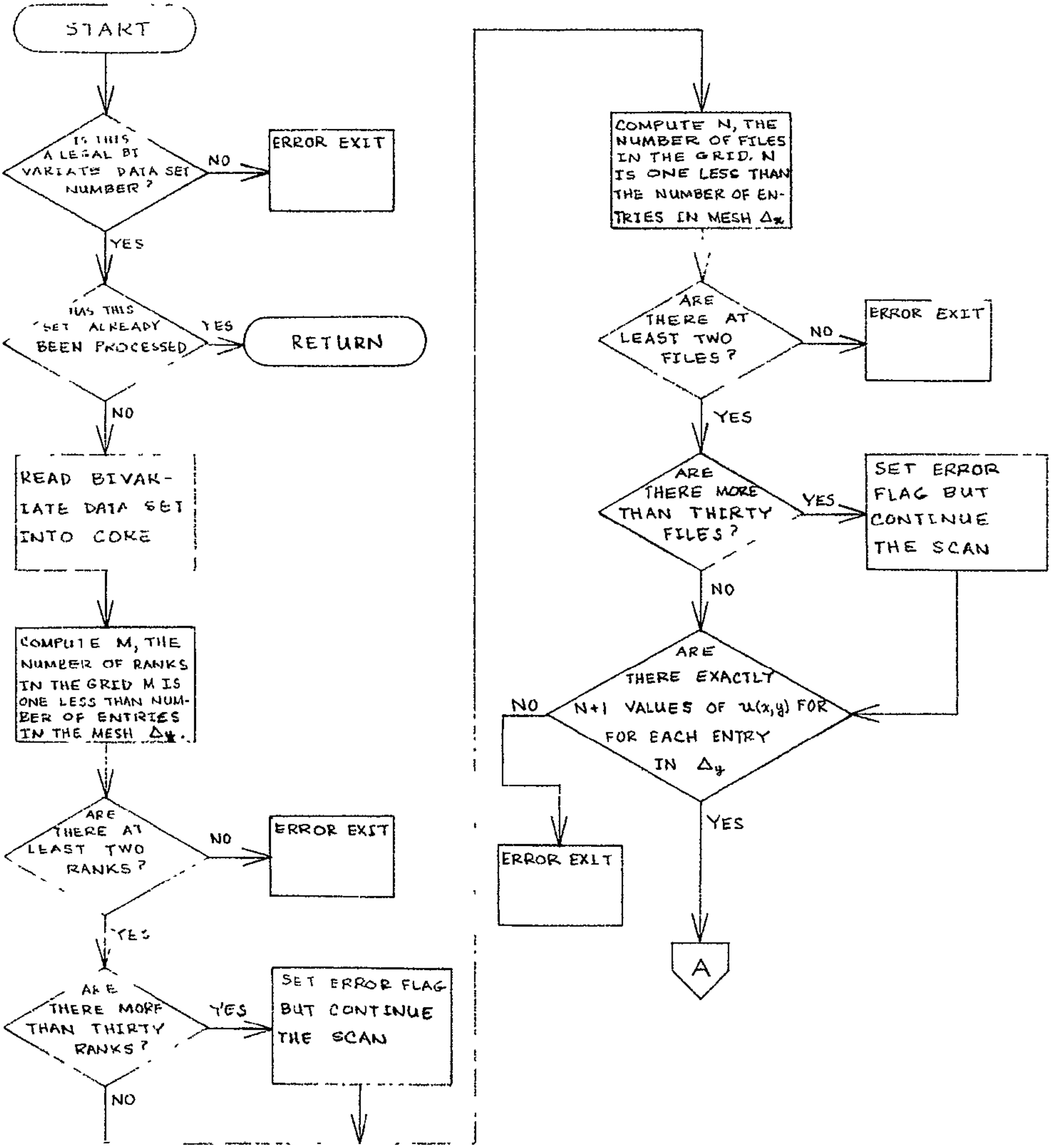
SUBRØUTINE
BLICØ

Subroutine BLICØ

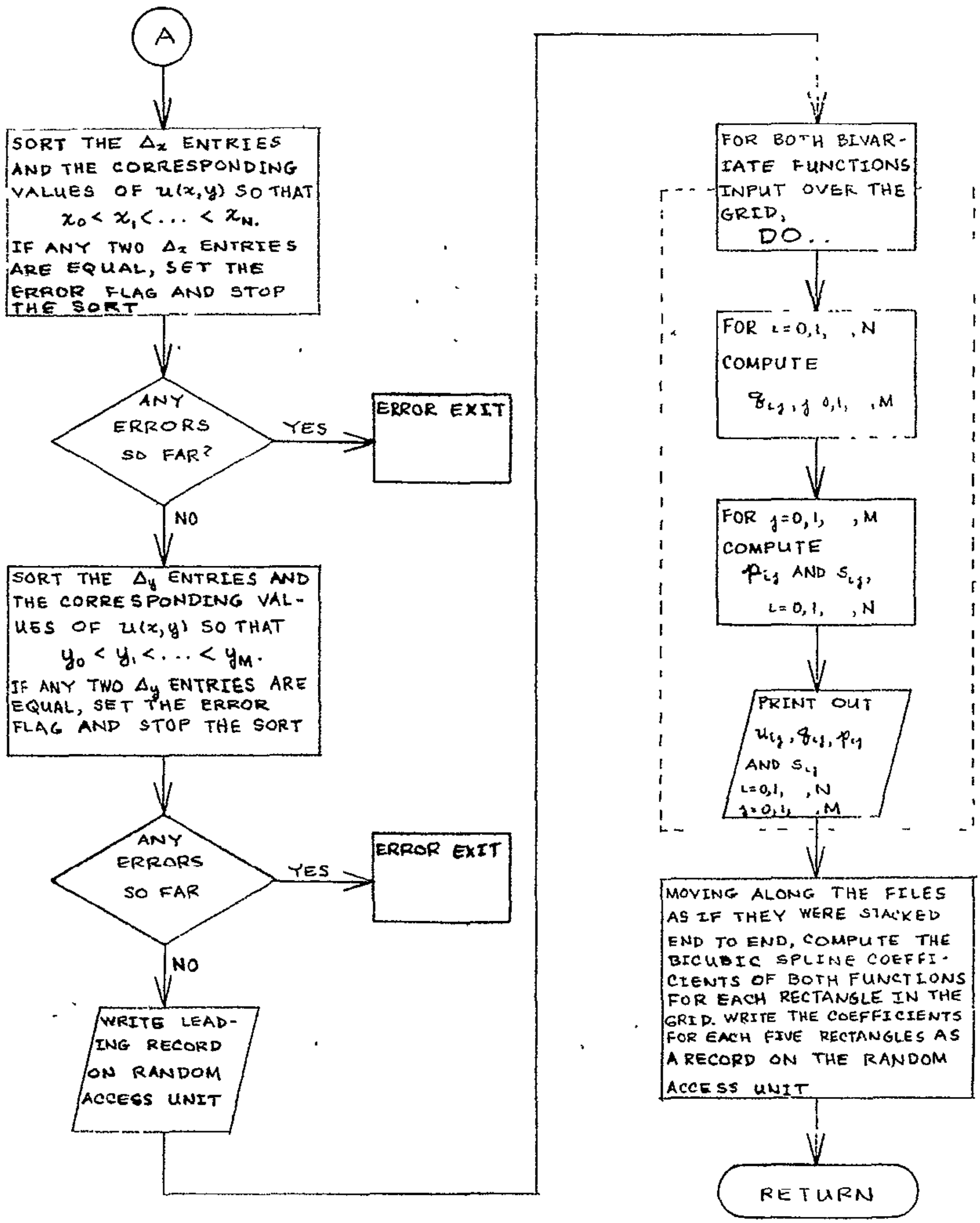
Purpose

Subroutine BLICØ computes bicube spline coefficients and stores them on disc or drum random files.

BLICØ


START
IS THIS A LEGAL BIVARIATE DATA SET NUMBER?
NO
ERROR EXIT
YES
HAS THIS SET ALREADY BEEN PROCESSED
YES
RETURN
NO
READ BIVARIATE DATA SET INTO CORE
COMPUTE M, THE NUMBER OF RANKS IN THE GRID M IS ONE LESS THAN NUMBER OF ENTRIES IN THE MESH Δ_y.
ARE THERE AT LEAST TWO RANKS?
NO
ERROR EXIT
YES
ARE THERE MORE THAN THIRTY RANKS?
YES
SET ERROR FLAG BUT CONTINUE THE SCAN
NO
COMPUTE N, THE NUMBER OF FILES IN THE GRID. N IS ONE LESS THAN THE NUMBER OF ENTRIES IN MESH Δ_x
ARE THERE AT LEAST TWO FILES?
NO
ERROR EXIT
YES
ARE THERE MORE THAN THIRTY FILES?
YES
SET ERROR FLAG BUT CONTINUE THE SCAN
NO
ARE THERE EXACTLY N+1 VALUES OF $u(x,y)$ FOR FOR EACH ENTRY IN Δ_y
NO
ERROR EXIT
YES
A

BLICØ (continued)



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ALFA		M	A 31 word array containing the mesh $y_0, y_1, \cdots, y_n$	/BICUBE/(	235)	BLICO	M	ALFA
						BLYNE	I	MACH
AMAX	y_m	O	The largest value of the second independent variable of a bivariate table.	/BICUBE/(	6)	BLICO	O	AMAX
						BLYNE	I	MMAX
AMIN	y_0	O	The smallest value of the second independent variable of a bivariate table.	/BICUBE/(	5)	BLICO	O	AMIN
						BLYNE	I	MMIN
F		I	A 31 word array containing the mesh $x_0, x_1, \cdots, x_n$	/BICUBE/(	204)	BLICO	I	F
						BLICO	M	MACH
						BLYNE	I	ALFA
ID		I	A four word array containing the basic deck, reference run, case and part case numbers in that order.	/GLOBAL/(	21)	BLICO	I	ID
						FRENCH	I	ID
						GEINP	I	ID
						PADS1	D	ID
						PRINT	I	ID
						SDINP	I	ID
						TOPM	I	ID
						VEHDF	I	ID
IF		M	Last rank in the grid in which interpolation occurred.	/BICUBE/(	7)	BLICO	M	IF
						BLYNE	M	IR
						INBVAD	M	IR
IFATAL		M	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADS1	I	IFATAL
						SDINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
IFMAX		M	Total number of ranks in grid.	/BICUBE/(	8)	BLICO	M	IFMAX
						BLYNE	I	IRMAX
						INBVAD	I	IRMAX
IREC		M	Logical record on IUNIT that contains spline coefficients for rectangle IRECT.	/BICUBE/(	11)	BLICO	M	IREC
						BLYNE	M	IREC
						INBVAD	M	IREC
IRECT		M	Grid rectangle associated with IR and IF.	/BICUBE/(	10)	BLICO	M	IRECT
						BLYNE	M	IRECT
						INBVAD	M	IRECT
IRMAX	N	M	Total number of files in grid.	/BICUBE/(	4)	BLICO	M	IRMAX
						BLYNE	I	IFMAX
						INBVAD	I	IFMAX
IUNIT		M	Logical unit number on which bicubic spline coefficients are stored for this table.	/BICUBE/(	9)	BLICO	M	IUNIT
						BLYNE	I	IUNIT
						INBVAD	I	IUNIT
MACH		M	A 31 word array containing the mesh $x_0, x_1, \cdot\ \ , x_n$	/BICUBE/(	204)	BLICO	I	F
						BLICO	M	MACH
						BLYNE	I	ALFA
MMAX	x_N	O	The largest value of the first independent variable of a bivariate table.	/BICUBE/(	2)	BLICO	O	MMAX
						BLYNE	I	AMAX
						INBVAD	I	AMAX
MMIN	x_0	M	The smallest value of the first independent variable of a bivariate table.	/BICUBE/(	1)	BLICO	M	MMIN
						BLYNE	I	AMIN
						INBVAD	I	AMIN
RDI		I	Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	D	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
T		1	A 160 word array containing logical record IREC	/BICUBE/(	44)	BLICO	I	T
						BLYNE	I	T
						INBVAD	I	T
.UN06.		0	File of all output data	/.UN06./(5	)	BLICO	0	UN06.
						BNDRYC	0	.UN06.
						CRASH	0	.UN06.
						FRENCH	0	.UN06.
						FXDAT	0	.UN06.
						GEINP	0	UN06.
						HUNT	0	.UN06.
						INEDIT	0	.UN06.
						ITER8	0	.UN06.
						MODELA	0	.UN06.
						MOMJ	0	.UN06.
						MPSI	0	.UN06.
						OUT	0	.UN06.
						PAY02	0	.UN06.
						PRINT	0	.UN06.
						PRINTV	0	.UN06.
						PRINTW	0	.UN06.
						PRITEQ	0	.UN06.
						PRITVA	0	.UN06.
						PROPIN	0	.UN06.
						PROTHR	0	.UN06.
						PRWTSM	0	.UN06.
						RANGE	0	.UN06.
						S	0	.UN06.
						SDINP	0	.UN06.
						SIZE	0	.UN06.
						SIZIN	0	.UN06.
						SIZOUT	0	UN06.
						SOLVE	0	.UN06.
						SPLICO	0	.UN06.
						SPLIZ	0	.UN06.
						SPLYNE	0	.UN06.
						SSSP	0	.UN06.
						STAU	0	.UN06.
						STPIT	0	.UN06.
						SUMOUT	0	.UN06.
						TABIN	0	UN06.
						TEST	0	.UN06.
						VEHDF	0	.UN06.
						WTSCH	0	UN06.
						WTVOL	0	.UN06.

```
      SUBROUTINE BLICO(ISET)                                            BLICO
      LOGICAL IFATAL, DONE(1), UNEQUL, RECYC                            BLICO
      REAL MACH(1), MSAVE, MMIN, MMAX                                   BLICO
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      ,ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      ,ARCDAT
     *XLMAX     ,HOMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      ,ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      ,ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      ,ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        ,ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                 ARCDAT
     * ,FRATE   , ARCD(9)                                               RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      COMMON/DATA/                                                      DATA
     *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      ,DATA
     *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4      DATA
      COMMON /GLOBAL/ G(1)                                              BLICO
      COMMON /TBLE/ U(1)                                                BLICO
      COMMON /BICUBE/ MMIN, MMAX, IR, IRMAX, AMIN, AMAX, IF, IFMAX,     BLICO
     1IUNIT, IRECT, IREC, C(32), T(160), F(1)                           BLICO
      COMMON /LASTAB/ LASTAB                                            BLICO
      DIMENSION ID(4)                                                   BLICO
      EQUIVALENCE(G(21),ID)                                             BLICO
      DIMENSION ALFA(1), IU(1), P(1922), Q(1922), S(1922), AY(16),      BLICO
     1AX(16), XK(16), TEMP(16)                                          BLICO
      EQUIVALENCE (IFATAL, G(17)), (DONE, G(12)), (IU, U),              BLICO
     1(ALFA, F(32)),               (MACH, F), (AY5, AY(5)), (AY6, AY(6)), BLICO
     2(AY7, AY(7)), (AY8, AY(8)), (AY14, AY(14)), (AY16, AY(16)),       BLICO
     3(AX2, AX(2)), (AX6, AX(6)), (AX8, AX(8)), (AX10, AX(10)),         BLICO
     4(AX14, AX(14)), (AX16, AX(16)), (XK1, XK(1)), (XK2, XK(2)),       BLICO
     5(XK3, XK(3)), (XK4, XK(4)), (XK5, XK(5)), (XK6, XK(6)),           BLICO
     6(XK7, XK(7)), (XK8, XK(8)), (XK9, XK(9)), (XK10, XK(10)),         BLICO
     7(XK11, XK(11)), (XK12, XK(12)), (XK13, XK(13)), (XK14, XK(14)),   BLICO
     8(XK15, XK(15)), (XK16, XK(16))                                    BLICO
      DATA AX/1., 5*0., .5, 9*0./,                                      BLICO
     1AY/1., 8*0., .5, 6*0./                                            BLICO
   10 FORMAT(1H0,  9HTHE VALUE, I3,49H IS AN ILLEGAL DATA  SET FOR BIVAR FIXED
     *IATE DATA       )                                                 FIXED
   20 FORMAT(1H0, 39HTHE BIVARIATE        DATA DATA  SET NO., I3,14H WAS FIXED
     *NOT INPUT)                                                        FIXED
   30 FORMAT(1H0, 66HLESS THAN 3 TABLES OF BIVARIATE       DATA WERE INP FIXED
     *UT IN DATA SET , I3)                                              FIXED
   40 FORMAT(1H0, 67HMORE THAN 31 TABLES OF BIVARIATE       DATA WERE IN FIXED
     *PUT IN DATA  SET, I3)                                             FIXED
   50 FORMAT(1H0, 77HLESS THAN 6 ENTRIES PER TABLE OF BIVARIATE       DA FIXED
     *TA WERE INPUT IN DATA  SET, I3)                                   FIXED
   60 FORMAT(1H0, 78HMORE THAN 62 ENTRIES PER TABLE OF BIVARIATE       D FIXED
     *ATA WERE INPUT IN DATA  SET, I3)                                  FIXED
   70 FORMAT(1H0, 26HBIVARIATE          DATA TABLE, I3, 13H IN DATA  SET, FIXED
     *I3, 13H IS TOO SHORT)                                             FIXED
   80 FORMAT(1H0, 26HBIVARIATE          DATA TABLE, I3, 13H IN DATA  SET, FIXED
     *I3, 12H IS TOO LONG)                                              FIXED
   90 FORMAT(1H0,  7HARG 1 =, E15.7, 27H IS NOT UNIQUE IN DATA  SET, I3) FIXED
  100 FORMAT(1H0, 6HARG 2=, E15.7, 27H IS NOT UNIQUE IN DATA  SET, I3)  FIXED
  110 FORMAT(1H0, 31HBIVARIATE  DATA       (DATA  SET, I3,  9H) FOLLOWS) FIXED
  120 FORMAT(1H1, 16HFIRST FUNCTION  )                                  FIXED
  130 FORMAT(1H1, 16HSECOND FUNCTION )                                  FIXED
  140 FORMAT(1H0, 8X, 7HARG 1 =, 8E14.6)                                FIXED
  150 FORMAT(1H0, 5HARG2=, F10.5, 8E14.6)                               FIXED
  160 FORMAT(1H , 15X, 8E14.6)                                          BLICO
  170 FORMAT(1H0, 27H*****FATAL INPUT ERROR*****)                       BLICO
      IF(6.LE.ISET.AND ISET.LE.9) GO TO 180                             FIXED   180
      IFATAL = .TRUE.                                                   BLICO
      WRITE(6,170)                                                      BLICO
      WRITE(6,10) ISET                                                  BLICO
      RETURN                                                            BLICO
  180 IF(DONE(ISET - 5)) RETURN                                         BLICO
      DONE(ISET - 5) = .TRUE.                                           BLICO
      CALL TABIN(DUMMY, 1, F, 70, U,1954, ID, ISET    ,0, IEOD)         BLICO
      IF(IEOD .EQ. 0) GO TO 190                                         BLICO   190
      IFATAL = .TRUE.                                                   BLICO
      WRITE(6,170)                                                      BLICO
      WRITE(6,20) ISET                                                  BLICO
      RETURN                                                            BLICO
```

```
      190 NT = IU - 1                                              BLICO
          IFMAX = NT - 1                                           BLICO
          IF(2 .LE. IFMAX) GO TO 200                               BLICO
          IFATAL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,30) ISET                                         BLICO
          RETURN                                                   BLICO
      200 IF(IFMAX .LE. 30) GO TO 210                              BLICO
          IFATAL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,40) ISET                                         BLICO
      210 NENT = LASTAB - 1                                        BLICO
          NWPT = NENT/NT                                           BLICO
          IRMAX = NWPT/2 - 1                                       BLICO
          NPPT = IRMAX + 1                                         BLICO
          IF(2 .LE  IRMAX) GO TO 220                               BLICO
          IFATAL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,50) ISET                                         BLICO
          RETURN                                                   BLICO
      220 IF(IRMAX .LE. 30) GO TO 230                              BLICO
          IFATAL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,60) ISET                                         BLICO
      230 UNEQUL = .FALSE.                                         BLICO
          DO 260 I = 1, IFMAX                                      BLICO
          IF(IU(I + 2) - IU(I + 1) - NWPT) 240,260,250             BLICO
      240 UNEQUL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,70) I, ISET                                      BLICO
          GO TO 260                                                BLICO
      250 UNEQUL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,80) I, ISET                                      BLICO
      260 CONTINUE                                                 BLICO
          IF(LASTAB - IU(IU) - NWPT) 270,290,280                   BLICO
      270 UNEQUL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,70) NT, ISET                                     BLICO
          GO TO 290                                                BLICO
      280 UNEQUL = .TRUE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,80) NT, ISET                                     BLICO
      290 IF(.NOT.UNEQUL) GO TO 300                                BLICO
          IFATAL = .TRUE.                                          BLICO
          RETURN                                                   BLICO
      300 RECYC = .FALSE.                                          BLICO
          DO 340 I = 1, IRMAX                                      BLICO
          IF(MACH(I + 1) - MACH(I)) 310,330,340                    BLICO
      310 IF(IFATAL) GO TO 340                                     BLICO
          RECYC =  TRUE.                                           BLICO
          MSAVE = MACH(I)                                          BLICO
          MACH(I) = MACH(I + 1)                                    BLICO
          MACH(I + 1) = MSAVE                                      BLICO
          DO 320 J = 1, NT                                         BLICO
          L = IU(J + 1) + 2*I + NT                                 BLICO
          CLSAVE = U(L - 1)                                        BLICO
          CDSAVE = U(L)                                            BLICO
          U(L - 1) = U(L + 1)                                      BLICO
          U(L) = U(L + 2)                                          BLICO
          U(L + 1) = CLSAVE                                        BLICO
      320 U(L + 2) = CDSAVE                                        BLICO
          GO TO 340                                                BLICO
      330 IFATAL = .TRUE.                                          BLICO
          RECYC =  FALSE.                                          BLICO
          WRITE(6,170)                                             BLICO
          WRITE(6,90) MACH(I), ISET                                BLICO
      340 CONTINUE                                                 BLICO
```

20 OCT 72 8.01-46

```
      IF(RECYC) GO TO 300                                      BLICO
      IF(IFATAL) RETURN                                        BLICO
  350 RECYC = .FALSE.                                          BLICO
      DO 390 I = 1, IFMAX                                      BLICO
      IF(ALFA(I + 1) - ALFA(I)) 360,380,390                    BLICO
  360 IF(IFATAL) GO TO 390                                     BLICO
      RECYC = .TRUE.                                           BLICO
      ASAVE = ALFA(I)                                          BLICO
      ALFA(I) = ALFA(I + 1)                                    BLICO
      ALFA(I + 1) = ASAVE                                      BLICO
      IBASE = IU(I + 1) + NT                                   BLICO
      DO 370 J = 1, NWPT, 2                                    BLICO
      K = J + IBASE                                            BLICO
      L = K + NWPT                                             BLICO
      CLSAVE = U(K)                                            BLICO
      CDSAVE = U(K + 1)                                        BLICO
      U(K) = U(L)                                              BLICO
      U(K + 1) = U(L + 1)                                      BLICO
      U(L) = CLSAVE                                            BLICO
  370 U(L + 1) = CDSAVE                                        BLICO
      GO TO 390                                                BLICO
  380 IFATAL = .TRUE.                                          BLICO
      RECYC = .FALSE.                                          BLICO
      WRITE(6,170)                                             BLICO
      WRITE(6,100) ALFA(I), ISET                               BLICO
  390 CONTINUE                                                 BLICO
      IF( RECYC) GO TO 350                                     BLICO
      IF(IFATAL) RETURN                                        BLICO
      IF(NWDB .GE. 0) GO TO 399                                JULY28
      L = NT                                                   SEP18
      NANT = NENT + NT + 1                                     SEP18
      DO 398 I = 1, NPPT                                       JULY28
      L = L + 2                                                JULY28
      AL = MACH(I)*RDI                                         JULY28
      SINAL = SIN(AL)                                          JULY28
      COSAL = COS(AL)                                          JULY28
      DO 397 J = L, NANT, NWPT                                 SEP18
      CN = U(J)                                                JULY28
      CX = U(J+1)                                              JULY28
      U(J) = CN*COSAL - CX*SINAL                               JULY28
      U(J+1) = CN*SINAL + CX*COSAL                             JULY28
  397 CONTINUE                                                 JULY28
  398 CONTINUE                                                 JULY28
  399 CONTINUE                                                 JULY28
      AMIN = ALFA                                              BLICO
      AMAX = ALFA(NT)                                          BLICO
      MMIN = MACH                                              BLICO
      MMAX = MACH(NPPT)                                        BLICO
      IUNIT = ISET + 27                                        BLICO
      CALL WRITMS(IUNIT, MMIN, 273, 1)                         BLICO
      DO 400 I = 1, NENT                                       BLICO
      P(I) = 0.                                                BLICO
      Q(I) = 0.                                                BLICO
  400 S(I) = 0.                                                BLICO
      WRITE(6,110) ISET                                        BLICO
      DO 430 IC = 1, 2                                         BLICO
      DO 410 J = IC, NWPT, 2                                   BLICO
      K = IU + J                                               BLICO
  410 CALL MOMENT(ALFA, IFMAX, U(K), P(J), NWPT)               BLICO
      DO 420 J = IC, NENT, NWPT                                BLICO
      K = IU + J                                               BLICO
      CALL MOMENT(MACH, IRMAX, U(K), Q(J), 2)                  BLICO
  420 CALL MOMENT(MACH, IRMAX, P(J), S(J), 2)                  BLICO
      DO 430 I = 1, NPPT, 8                                    BLICO
      ILIM = MIN0(I + 7, NPPT)                                 BLICO
      II = IC - NWPT + 2*(I - 1)                               BLICO
      IK = IU + II                                             BLICO
      IIMAX = ILIM*2 - NWPT                                    BLICO
      IKMAX = IIMAX + IU                                       BLICO
      DO 430 J = 1, NT, 11                                     BLICO
      IF(IC .EQ. 1) WRITE(6,120)                               BLICO
      IF(IC .EQ. 2)WRITE(6,130)                                BLICO
      WRITE(6,140) (MACH(IJ), IJ = I, ILIM)                    BLICO
```

20 OCT 72 8.01-46

```
      JLIM = MINO(J + 10, NT)                                   BLICO
      DO 430 L = J, JLIM                                        BLICO
      IK = IK + NWPT                                            BLICO
      II = II + NWPT                                            BLICO
      IIMAX = IIMAX + NWPT                                      BLICO
      IKMAX = IKMAX + NWPT                                      BLICO
      WRITE(6,150) ALFA(L), (U(IJ), IJ = IK, IKMAX, 2)          BLICO
      WRITE(6,160) (P(IJ), IJ = II, IIMAX, 2)                   BLICO
      WRITE(6,160) (Q(IJ), IJ = II, IIMAX, 2)                   BLICO
  430 WRITE(6,160) (S(IJ), IJ = II, IIMAX, 2)                   BLICO
      L = 0                                                     BLICO
      NRECT = IRMAX*IFMAX                                       BLICO
      NREC = (NRECT + 4)/5 + 1                                  BLICO
      IRS = -1                                                  BLICO
      I = -4                                                    BLICO
      DO 460 IREC = 2, NREC                                     BLICO
      I = I + 5                                                 BLICO
      ILIM = MINO(I + 4, NRECT)                                 BLICO
      K = -15                                                   BLICO
      DO 450 IRECT = I, ILIM                                    BLICO
      IRM1 = (IRECT - 1)/IFMAX                                  BLICO
      IF = IRECT - IFMAX*IRM1                                   BLICO
      IF(IRM1 .EQ  IRS) GO TO 440                               BLICO  440
      X = MACH(IRM1 + 2) - MACH(IRM1 + 1)                       BLICO
      AX2 = -1./X                                               BLICO
      AX6 = -X/3.                                               BLICO
      AX8 = -1./(6.*X)                                          BLICO
      AX10= -AX2                                                BLICO
      AX14= AX6/2.                                              BLICO
      AX16= -AX8                                                BLICO
      IRS = IRM1                                                BLICO
  440 Y = ALFA(IF + 1) - ALFA(IF)                               BLICO
      AY5 = -1./Y                                               BLICO
      AY6 = -Y/3.                                               BLICO
      AY7 = -AY5                                                BLICO
      AY8 = AY6/2.                                              BLICO
      AY14= -1./(6 *Y)                                          BLICO
      AY16= -AY14                                               BLICO
      IXP = 2*IRM1 + NWPT*(IF - 1)                              BLICO
      DO 450 IC = 1, 2                                          BLICO
      K = K + 16                                                BLICO
      IX = IXP + IC                                             BLICO
      L = IX + IU                                               BLICO
      XK1 = U(L)                                                BLICO
      XK2 = Q(IX)                                               BLICO
      XK5 = P(IX)                                               BLICO
      XK6 = S(IX)                                               BLICO
      IX = IX + 2                                               BLICO
      L = IX + IU                                               BLICO
      XK3 = U(L)                                                BLICO
      XK4 = Q(IX)                                               BLICO
      XK7 = P(IX)                                               BLICO
      XK8 = S(IX)                                               BLICO
      IX = IX - 2 + NWPT                                        BLICO
      L = IX + IU                                               BLICO
      XK9 = U(L)                                                BLICO
      XK10 = Q(IX)                                              BLICO
      XK13 = P(IX)                                              BLICO
      XK14= S(IX)                                               BLICO
      IX = IX + 2                                               BLICO
      L = IX + IU                                               BLICO
      XK11 = U(L)                                               BLICO
      XK12 = Q(IX)                                              BLICO
      XK15 = P(IX)                                              BLICO
      XK16= S(IX)                                               BLICO
      CALL MATMLT(TEMP, XK, AY, 4, 4, 4)                        BLICO
  450 CALL MATMLT(T(K), AX, TEMP, 4, 4, 4)                      BLICO
  460 CALL WRITMS(IUNIT, T, 160, IREC)                          BLICO
      RETURN                                                    BLICO
      END                                                       BLICO
```

20 OCT 72 6 01-46

SUBRØUTINE BNDRYC

Subroutine BNDRYC

Purpose

This subroutine packs boundary condition data (both initial and target input data) into an ordered packed array which will be stored in a random access file.

Description

The input to this routine is the array ARR which is an unordered packed representation of the input initial and target conditions. The main function of the routine is to sort the unordered data into a matrix array, concurrently counting the numbers of initial and target conditions in each arc. The number of initial and target conditions is stored by the calling program (GEINP) in arrays ITAB and KTAB respectively.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ACD		W	Intermediate unpacked boundary condition array	/BNDRYC/(*	)	BNDRYC	W	ACD
ANC		W	Intermediate unpacked boundary condition array	/BNDRYC/(*	)	BNDRYC	W	ANC
ARR		I	Array of input initial and target conditions	/BNDRYC/(*	)	BNDRYC	I	ARR
BNARR		O	Packed boundary condition array output on random file	/BNDRYC/(*	)	BNDRYC	O	BNARR
ILOC		I	Index of array ARR where the boundary conditions start	/BNDRYC/(*	)	BNDRYC	I	ILOC
ITAB		M	Array of number of initial or target conditions per arc	/BNDRYC/(*	.)	BNDRYC	M	ITAB
M		W	Keep track of location in BNARR array	/BNDRYC/(*	)	BNDRYC	W	M
YZ		W	Array of unpacked boundary condition values	/BNDRYC/(*	)	BNDRYC	W	YZ
UN06.		O	File of all output data	/.UN06./($	)	BLICO	O	.UN06.
						BNDRYC	O	.UN06.
						CRASH	O	.UN06.
						FRENCH	O	UN06.
						FXDAT	O	.UN06.
						GEINP	O	.UN06.
						HUNT	O	.UN06.
						INEDIT	O	.UN06.
						ITER8	O	.UN06.
						MODELA	O	.UN06.
						MOMJ	O	.UN06.
						MPSI	O	.UN06.
						OUT	O	.UN06.
						PAYO2	O	.UN06.
						PRINT	O	.UN06.
						PRINTV	O	UN06.
						PRINTW	O	.UN06.
						PRITEQ	O	.UN06.
						PRITVA	O	.UN06.
						PROPIN	O	.UN06.
						PROTHR	O	.UN06.
						PRWTSM	O	.UN06.
						RANGE	O	.UN06.
						S	O	.UN06.
						SDINP	O	.UN06.
						SIZE	O	.UN06.
						SIZIN	O	.UN06.
						SIZOUT	O	UN06.
						SOLVE	O	.UN06.
						SPLICO	O	.UN06.
						SPLIZ	O	.UN06.
						SPLYNE	O	.UN06.
						SSSP	O	.UN06.
						STAU	O	.UN06.
						STPIT	O	.UN06.
						SUMOUT	O	.UN06.
						TABIN	O	.UN06.
						TEST	O	.UN06.
						VEHDF	O	.UN06.
						WTSCH	O	.UN06.
						WTVOL	O	.UN06.

```
C      SUBROUTINE  BNDRYC(ARR,ILOC,ITAB,BNARR, N,IEND,IFATAL)           BNDRY
C      ARR  IS  THE       INPUT ARRAY CONTAINING BNDRY COND INPUT        BNDRY
C      ILOC IS  THE  INDEX OF ARRAY IRR WHERE THE BNDRY COND. INPUT START BNDRY
C      ITAB IS  THE  ARRAY  WHOSE VALUES ARE THE  NUMBER OF  INITIAL     BNDRY
C        CONDITIONS SPECIFIED IN EACH ARC                                BNDRY
C      BNARR  IS  AN ARRAY  CONTAINING  IC  VALUES  AND  CLASSIFICATIONS BNDRY
C      FOR EXAMPLE CLASSIFICATION CODES ARE*                             BNDRY
C                      0.  CONTINUOUS                                    BNDRY
C                       1.  KNOWN                                        BNDRY
C                       2.  OPTIMIZED                                    BNDRY
C                       3.  ESTIMATED                                    BNDRY
C                       4.  UNKNOWN                                      BNDRY
C                      5. DISCONTINUITY IN  MASS (WEIGHT)                COMN
C                      6  WEIGHT DISCONTIUTY COMPUTED IN WTDROP          COMN
C                     15   BRANCH TO STAGE 5  (FOR EXAMPLE)              BNDRY
C                ***        ***        ***                               BNDRY
       LOGICAL IFATAL                                                    BNDRY
       DIMENSION ITAB(1),BNARR(1),ARR(1),ACD(10,20),YZ(10,20),ANC(10,20), BNDRY
      1AVA(10,20)                                                        BNDRY
       EQUIVALENCE (ACD,ANC),(AVA,YZ)                                    BNDRY
C                                                                        COMN
C         I    ZERO  ARRAYS                                              COMN
       DO 10  I=1,200                                                    BNDRY
    10 ACD(I)=0.                                                         BNDRY
       WRITE(6,20)                                                       BNDRY
    20 FORMAT (56H1               PRELIMINARY SCAN  OF  BOUNDARY CONDITION BNDRY
      1S)                                                                BNDRY
C                                                                        COMN
C         II   EXTRACT  IC  DATA FROM   ARR   AND LOAD INTO              COMN
C              MATRICES ,ALSO  COUNT  NO OF ICS  AND STORE IN ITAB       COMN
       KCNT =0                                                           BNDRY
       J=ILOC                                                            BNDRY
    30 IF(J.GT IEND) GO TO 50                                            BNDRY
       IA1= ARR(J)+.5                                                    BNDRY
       ICD1=ARR(J+1) +.5                                                 BNDRY
       IF(ARR(J+2).NE.0.)                                                BNDRY
      1ITAB(IA1)= ITAB(IA1)+1                                            BNDRY
       ACD(ICD1,IA1) =ARR(J+2)                                           BNDRY
       YZ (ICD1,IA1) =ARR(J+3)                                           BNDRY
       KCNT = KCNT +1                                                    BNDRY
    40 J =J +4                                                           BNDRY
       GO TO 30                                                          BNDRY
    50 CONTINUE                                                          BNDRY
C                                                                        COMN
C         III  CHECK FOR ICS IN EACH ARC  AND PUT CODES  IN BNARR        COMN
C       PACK  IC DATA IN  BNARR                                          BNDRY
       M=0                                                               BNDRY
       DO 100 I=1,20                                                     BNDRY
       IF(ITAB(I).EQ.0)GO TO 100                                         BNDRY
       JC=ITAB(I)                                                        BNDRY
       DO 90 K=1,10                                                      BNDRY
       IF(ACD(K,I).EQ.0) GO TO 90                                        BNDRY
    60 BNARR(M+1)= ACD(K,I)                                              BNDRY
    70 BNARR(M+3) = YZ(K,I)                                              BNDRY
    80 BNARR(M+2) = K                                                    BNDRY
       M=M+3                                                             BNDRY
C                                                                        COMN
C         III-A  TEST  FOR  TOO MANY  ICS                                COMN
       IF(M GT.N) GO TO 110                                              BNDRY
    90 CONTINUE                                                          BNDRY
   100 CONTINUE                                                          BNDRY
       GO TO 140                                                         BNDRY
   110 WRITE(6,130)                                                      BNDRY
       IFATAL= .TRUE.                                                    BNDRY
   120 WRITE(6,150)                                                      BNDRY
       RETURN                                                            BNDRY
   130 FORMAT(3X,33H TO MANY IC S SPECIFIED FOR ARRAY)                   BNDRY
   140 WRITE(6,160)                                                      BNDRY
   150 FORMAT(39H   BAD SCAN OF INPUT INITIAL CONDITIONS)                BNDRY
   160 FORMAT(39H  GOOD SCAN OF INPUT INITIAL CONDITIONS)                BNDRY
       RETURN                                                            BNDRY
C         IV     TARGET  CONDITION  SCAN  AND  PACKING                   COMN
```

```
C                                                                      COMN
C     ARR  IS THE INPUT ARRAY                                          BNDRY
C     ILOC IS THE LOCATION IN ARR WHERE TARGET INPUTS START            BNDRY
C     KTAB IS THE ARRAY OF NO OF TARGETS PER  ARC                      BNDRY
C     TARG IS  ARRAY OF TARGETS AND  TARGET CLASSIFICATIONS            BNDRY
C     IEND IS  LAST  WORD IN ARR  ASSOCIATED  WITH TARGETS             BNDRY
      ENTRY TARGCM                                                     BNDRY
C                                                                      COMN
C    IV-A ZERO  MATRICES                                               COMN
      DO 170 I=1,200                                                   BNDRY
  170 ACO(I)=0.                                                        BNDRY
C                                                                      COMN
C    IV-B   EXTRACT DATA FROM INPUT ARRAY AND PUT IN MATRICES          COMN
C             ALSO  COUNT  NO OF TARGETS IN EACH ARC                   COMN
      KCNT =0                                                          BNDRY
      J=ILOC                                                           BNDRY
  180 IF(J.GT IEND) GO TO 190                                          BNDRY
      IA1=  ARR(J)+.5                                                  BNDRY
      ITAB(IA1) = ITAB(IA1)+1                                          BNDRY
      KK= ITAB(IA1)                                                    BNDRY
      ANC(KK,IA1) =  (ARR(J+1)*1.E6 + ABS(ARR(J+2)))*SIGN (1.,ARR(J+2)) BNDRY
      AVA(KK,IA1) =  ARR(J+3)                                          BNDRY
      J= J+4                                                           BNDRY
      GO TO 180                                                        BNDRY
  190 CONTINUE                                                         BNDRY
C                                                                      COMN
C     V    PACK  DATA  IN  TARG  ARRAY                                 COMN
      M=1                                                              BNDRY
      DO 210 I=1,20                                                    BNDRY
      IF(ITAB(I).EQ.0) GO TO 210                                       BNDRY
      K= ITAB(I)                                                       BNDRY
      DO 200  JK=1,K                                                   BNDRY
      BNARR(M) = ANC(JK,I)                                             BNDRY
      BNARR(M+1)=AVA(JK,I )                                            BNDRY
      M=M+2                                                            BNDRY
  200 CONTINUE                                                         BNDRY
  210 CONTINUE                                                         BNDRY
      WRITE(6,220)                                                     BNDRY
  220 FORMAT(32H  GOOD SCAN OF TARGET CONDITIONS)                      BNDRY
      RETURN                                                           BNDRY
      END                                                              BNDRY
```

20 OCT 72 6.01-46

SUBROUTINE MOMENT

```
      SUBROUTINE MOMENT(X, ILIM, Y, Z, KINK)                            MOMENT
      DIMENSION X(1), Y(1), Z(1), Q(50), U(50)                          MOMENT
      Q = 0.                                                            MOMENT
      U = 0.                                                            MOMENT
      K = 1                                                             MOMENT
      KP1 = 1 + KINK                                                    MOMENT
      DO 10 I = 2, ILIM                                                 MOMENT
      IM1 = I - 1                                                       MOMENT
      KM1 = K                                                           MOMENT
      K = KP1                                                           MOMENT
      KP1 = KP1 + KINK                                                  MOMENT
      XI = X(I)                                                         MOMENT
      YK = Y(K)                                                         MOMENT
      H1 =  XI - X(IM1)                                                 MOMENT
      H2 =  X(I + 1) - XI                                               MOMENT
      H1PH2 = H1 + H2                                                   MOMENT
      XLAM = H2/H1PH2                                                   MOMENT
      XMU = 1. - XLAM                                                   MOMENT
      D =6./H1PH2*((Y(KP1) - YK)/H2 - (YK - Y(KM1))/H1)                 MOMENT
      P = XMU*Q(IM1) + 2.                                               MOMENT
      Q(I) = -XLAM/P                                                    MOMENT
   10 U(I) = (D - XMU*U(IM1))/P                                         MOMENT
      L = ILIM                                                          MOMENT
      DO 20 I = 2, ILIM                                                 MOMENT
      Z(K) = Q(L)*Z(KP1) + U(L)                                         MOMENT
      KP1 = K                                                           MOMENT
      K = K - KINK                                                      MOMENT
   20 L = L - 1                                                         MOMENT
      RETURN                                                            MOMENT
      END                                                               MOMENT
```

20 OCT 72 6.01-46

SUBRØUTINE SPLICØ

Subroutine SPLICØ

Purpose

Subroutine SPLICØ computes spline fit coefficients for univariant tables.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IFATAL		M	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADSI	I	IFATAL
						SOINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
LOCF		M	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the last value of the independent variable of the corresponding table.	/TABLE /(	1401)	SPLICO	M	LOCF
						SPLICO	M	Z
						SPLIZ	I	LOCF
						SPLIZ	I	Z
						SPLYNE	I	LOCF
						SPLYNE	I	Z
						THRUP	O	Z
LOCI		M	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that points to the initial value of the independent variable of the corresponding table A zero entry indicates table not input.	/TABLE /(	1)	SPLICO	M	LOCI
						SPLICO	M	X
						SPLIZ	I	LOCI
						SPLIZ	I	X
						SPLYNE	I	LOCI
						SPLYNE	I	X
						THRUP	I	LOCI
						THRUP	O	X
LOCL		O	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that indicates the last interval in which interpolation of the corresponding table occured.	/TABLE /(	701)	SPLICO	O	LOCL
						SPLICO	M	Y
						SPLIZ	M	LOCL
						SPLIZ	I	Y
						SPLYNE	M	LOCL
						SPLYNE	I	Y
						THRUP	O	Y
NT		M	Largest univariant table number in this case.	/GLOBAL/(	66)	SPLICO	M	NT
						SPLIZ	I	NT
						SPLYNE	I	NT
X		M	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the initial value of the independent variable of the corresponding table. A zero entry indicates table not input.	/TABLE /(	1)	SPLICO	M	LOCI
						SPLICO	M	X
						SPLIZ	I	LOCI
						SPLIZ	I	X
						SPLYNE	I	LOCI
						SPLYNE	I	X
						THRUP	I	LOCI
						THRUP	O	X
Y		M	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that indicates the last interval in which interpolation of the corresponding table occured.	/TABLE /(	701)	SPLICO	O	LOCL
						SPLICO	M	Y
						SPLIZ	M	LOCL
						SPLIZ	I	Y
						SPLYNE	M	LOCL
						SPLYNE	I	Y
						THRUP	O	Y
Z		M	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that points to the last value of the independent variable of the corresponding table.	/TABLE /(	1401)	SPLICO	M	LOCF
						SPLICO	M	Z
						SPLIZ	I	LOCF
						SPLIZ	I	Z
						SPLYNE	I	LOCF
						SPLYNE	I	Z
						THRUP	O	Z

153

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
UN06		0	File of all output data	/.UN06./($	)	BLICO	0	.UN06.
						BNDRYC	0	UN06.
						CRASH	0	.UN06.
						FRENCH	0	.UN06.
						FXDAT	0	.UN06.
						GEINP	0	UN06
						HUNT	0	UN06.
						INEDIT	0	.UN06
						ITER8	0	UN06.
						MODELA	0	.UN06.
						MOMJ	0	.UN06.
						MPSI	0	.UN06.
						OUT	0	.UN06.
						PAYO2	0	.UN06.
						PRINT	0	.UN06.
						PRINTV	0	.UN06.
						PRINTW	0	UN06.
						PRITEQ	0	.UN06.
						PRITVA	0	UN06.
						PROPIN	0	.UN06.
						PROTHR	0	.UN06.
						PRWTSM	0	.UN06.
						RANGE	0	UN06.
						S	0	.UN06.
						SDINP	0	.UN06.
						SIZE	0	.UN06.
						SIZIN	0	.UN06.
						SIZOUT	0	.UN06.
						SOLVE	0	.UN06.
						SPLICO	0	.UN06.
						SPLIZ	0	.UN06.
						SPLYNE	0	.UN06.
						SSSP	0	.UN06.
						STAU	0	.UN06.
						STPIT	0	.UN06.
						SUMOUT	0	.UN06
						TABIN	0	.UN06.
						TEST	0	UN06.
						VEHDF	0	.UN06.
						WTSCH	0	.UN06.
						WTVOL	0	.UN06.

154


```
      SUBROUTINE SPLICO                                                 SPLICO
      LOGICAL IFATAL                                                    SPLICO
      COMMON /GLOBAL/G(66) /TABLE/T(2100) /TBLE/TBLE(1) /LASTAB/LASTAB  SPLICO
      DIMENSION ITBLE(1), LOCI(1), LOCL(1), LOCF(1), X(1), Y(1), Z(1)   SPLICO
      EQUIVALENCE (ITBLE, TBLE), (IFATAL, G(17)), (LOCI, T), (LOCL, T(70 SPLICO
     11)), (LOCF, T(1401)), (X, LOCI), (Y, LOCL), (Z, LOCF), (NT, G(66)) SPLICO
   10 FORMAT(1H0, 18H FATAL INPUT ERROR)                                 SPLICO
   20 FORMAT(1H0,  77H TOO MANY TABLES AND/OR TOO MANY POINTS.  THE SUM SPLICO
     1OF THE LARGEST TABLE NUMBER/ 77H  AND THE NUMBER OF POINTS IN ALL  SPLICO
     2OF THE TABLES MUST BE LESS THAN OR EQUAL TO/ 54H  700.  IN THIS IN SPLICO
     3STANCE, THE LARGEST TABLE NUMBER IS , I3, 19H, AND THE NUMBER OF/  SPLICO
     433H  POINTS IN ALL OF THE TABLES IS , I3, 1H.)                     SPLICO
   30 FORMAT(1H0, 13H IN TABLE NO., I3,  14H, THE POINT X=, E14.7, 17H I SPLICO
     1S NOT DISTINCT.)                                                   SPLICO
   40 FORMAT(1H0, 26X, 9HTABLE NO., I3//1H , 2HX=,E14 7, 5X, 5HF(X)=,   SPLICO
     1E14.7, 5X, 5HM(X)=,E14 7/(1H , 2X, E14.7, 2(10X, E14.7)))          SPLICO
      NT = ITBLE - 1                                                    SPLICO
      IF(NT.GT.0) GO TO 45                                              SEP18
      NT = 0                                                            SEP18
      RETURN                                                            SEP18
   45 CONTINUE                                                          SEP18
      NP = (LASTAB - 1)/2                                               SPLICO
      LAST = NT + NP                                                    SPLICO
      IF(LAST .LE. 700) GO TO 50                                        SPLICO
      IFATAL = .TRUE.                                                   SPLICO
      WRITE(6,10)                                                       SPLICO
      WRITE(6,20) NT, NP                                                SPLICO
      RETURN                                                            SPLICO
   50 DO 60 I = 1, NT                                                   SPLICO
      LOCI(I) = 0                                                       SPLICO
      LOCL(I) = 0                                                       SPLICO
   60 LOCF(I) = 0                                                       SPLICO
      J = NT + 1                                                        SPLICO
      K = NT                                                            SPLICO
      DO 70 I = J, LAST                                                 SPLICO
      K = K + 2                                                         SPLICO
      X(I) = TBLE(K)                                                    SPLICO
      Y(I) = TBLE(K + 1)                                                SPLICO
   70 Z(I) = 0.                                                         SPLICO
      LAST = NT                                                         SPLICO
      NEXT = LAST + 1                                                   SPLICO
      DO 110 I = 1, NT                                                  SPLICO
      K = I + 1                                                         SPLICO
      LOCO = ITBLE(K)                                                   SPLICO
      IF(LOCO .EQ  0) GO TO 110                                         SPLICO
      LOCI(I) = NEXT                                                    SPLICO
      LOCL(I) = NEXT                                                    SPLICO
      IF(I .LT. NT) GO TO 80                                            SPLICO
      LAST = NT + NP                                                    SPLICO
      GO TO 100                                                         SPLICO
   80 DO 90 J = K, NT                                                   SPLICO
      LOCA = ITBLE(J + 1)                                               SPLICO
      IF(LOCA  EQ  0) GO TO 90                                          SPLICO
      LAST = LAST + (LOCA - LOCO)/2                                     SPLICO
      GO TO 100                                                         SPLICO
   90 CONTINUE                                                          SPLICO
  100 LOCF(I) = LAST                                                    SPLICO
      NEXT = LAST + 1                                                   SPLICO
  110 CONTINUE                                                          SPLICO
      DO 170 IT = 1, NT                                                 SPLICO
      II = LOCI(IT)                                                     SPLICO
      IF(II .EQ. 0) GO TO 170                                           SPLICO
      IF = LOCF(IT)                                                     SPLICO
      IF(II .EQ. IF) GO TO 160                                          SPLICO
      IIP1 = II + 1                                                     SPLICO
  120 IRECYC = 0                                                        SPLICO
      DO 150 I = IIP1, IF                                               SPLICO
      IM1 = I - 1                                                       SPLICO
      XS = X(IM1)                                                       SPLICO
      XI = X(I)                                                         SPLICO
      IF(XI - XS) 130,140,150                                           SPLICO
```

20 OCT 72 8.01-46

135/

```
  130 YS = Y(IM1)                                        SPLICO
      X(IM1) = XI                                        SPLICO
      Y(IM1) = Y(I)                                      SPLICO
      X(I) = XS                                          SPLICO
      Y(I) = YS                                          SPLICO
      IF( NOT.IFATAL) IRECYC = 1                         SPLICO
      GO TO 150                                          SPLICO   150
  140 IFATAL =  TRUE.                                    SPLICO
      IRECYC = 0                                         SPLICO
      WRITE(6,10)                                        SPLICO
      WRITE(6,30) IT, XI                                 SPLICO
  150 CONTINUE                                           SPLICO
      IF(IRECYC .GT  0) GO TO 120                        SPLICO   120
      IF(IFATAL) GO TO 160                               SPLICO   160
      IFMII= IF - II                                     SPLICO
      IF(IFMII  LT  2)   GO TO 160                       SPLICO   160
      CALL MOMENT(X(II), IF M II, Y(II), Z(II), 1)       SPLICO
  160 WRITE(6,40) IT, (X(IJ), Y(IJ), Z(IJ), IJ = II, IF) SPLICO
  170 CONTINUE                                           SPLICO
      RETURN                                             SPLICO
      END                                                SPLICO
```

20 OCT 72 6 01-46

156

161

Steepest Descent Trajectory Module (TØPM) 5

CONTENTS

Subroutine	BSTØ3
Subroutine	CØN3
Subroutine	DER3A
Subroutine	DTF3
Subroutine	EL1
Subroutine	EL2
Subroutine	ENVPRM
Subroutine	EQUA3
Subroutines	FH1 → FH4
Subroutine	FNTG
Subroutine	FXDAT
Subroutine	GETIT
Subroutine	GUI3A
Function	ICOD
Subroutine	IMPUL
Subroutine	IPR
Function	LØMG
Subroutine	MAMECØ
Subroutine	MØDELA
Subroutine	MØDELB
Function	MØMJ
Function	MPSI
Subroutine	MTX3A
Subroutine	ØUT
Subroutine	PAT63
Subroutine	PAYØ2
Subroutine	PDBC
Subroutine	PDY3A
Subroutine	PRMSET
Subroutine	PRØPB
Subroutine	PRØPIN
Subroutine	REU3
Subroutine	RKTA3A
Subroutine	RKTB3A

Subroutine	SDER3
Subroutine	SDINP
Function	SØMG
Subroutine	SPLYNE
Subroutine	STAU
Subroutine	STPIT
Subroutine	STP3
Subroutine	STVRL3
Function	SUMS
Subroutine	SYMVRT
Subroutine	TBLK
Subroutine	TEST
Function	TØLPSI
Function	TØL3
Subroutine	TRAN3
Subroutine	TRTØSZ
Subroutine	VT
Subroutine	YREF3

SUBRØUTINE
TØPM

TOPM

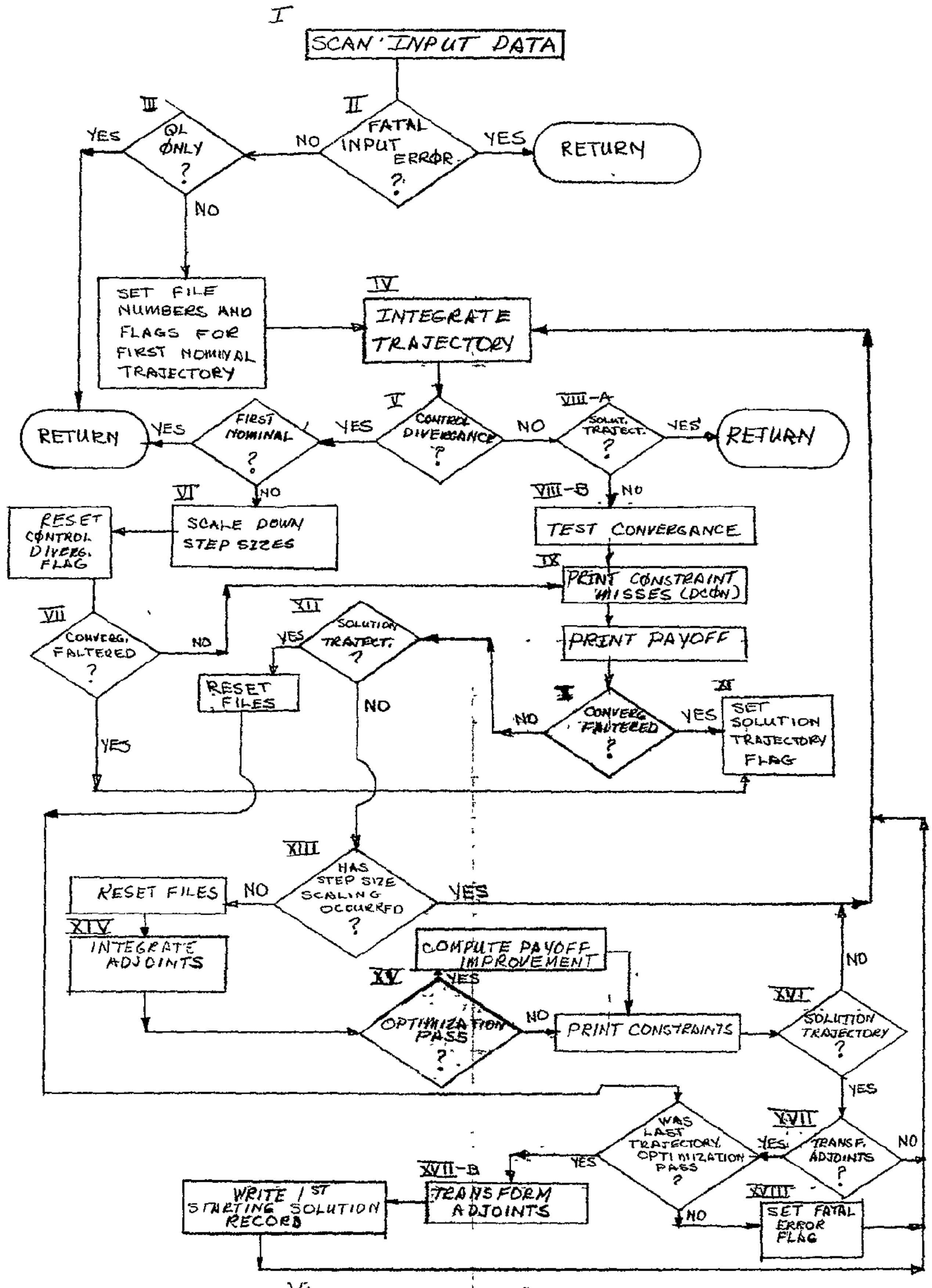
Purpose

TOPM is the Steepest Descent module main executive program.

Description

The reader is referred to Section 2.0 of Volume II for a chart of the overlay structure particularly under this routine (TOPM). The chart shows the names of the routines that are called on UNIVAC equipment instead of the overlay numbers used on CDC equipment. A further description of the iteration procedure generated by TOPM is contained in Section 15.3 of Volume I.

TOPM



I
SCAN INPUT DATA
II
FATAL INPUT ERROR ?
YES
RETURN
NO
III
QL ONLY ?
YES
NO
SET FILE NUMBERS AND FLAGS FOR FIRST NOMINAL TRAJECTORY
IV
INTEGRATE TRAJECTORY
V
CONTROL DIVERGANCE ?
YES
FIRST NOMINAL ?
YES
RETURN
NO
VI
SCALE DOWN STEP SIZES
RESET CONTROL DIVERG. FLAG
VII
CONVERG. FALTERED ?
NO
YES
NO
VIII-A
SOLUT. TRAJECT. ?
YES
RETURN
NO
VIII-B
TEST CONVERGANCE
IX
PRINT CONSTRAINT MISSES (DCON)
PRINT PAYOFF
X
CONVERG. FALTERED ?
NO
YES
XI
SET SOLUTION TRAJECTORY FLAG
XII
SOLUTION TRAJECT. ?
YES
RESET FILES
NO
XIII
HAS STEP SIZE SCALING OCCURRED ?
NO
RESET FILES
YES
XIV
INTEGRATE ADJOINTS
XV
OPTIMIZATION PASS ?
YES
COMPUTE PAYOFF IMPROVEMENT
NO
PRINT CONSTRAINTS
XVI
SOLUTION TRAJECTORY ?
NO
YES
XVII
TRANSF. ADJOINTS ?
NO
YES
WAS LAST TRAJECTORY OPTIMIZATION PASS ?
YES
NO
XVII-B
TRANSFORM ADJOINTS
WRITE 1ST STARTING SOLUTION RECORD
XVIII
SET FATAL ERROR FLAG

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
BCON		I	Vector of constraint misses on trial trajectory	/GENF /(	199)	CON3	O	BCON
						TEST	I	BCON
						TOPM	I	BCON
DCON	$d\Psi_i$	I	Asked for correction in constraint misses and payoff vector	/GENF /(	289)	CON3	O	DCON
						MTX3A	I	DCON
						PAYO2	M	DCON
						TEST	M	DCON
						TOPM	I	DCON
						TRTOSZ	I	DCON
DELP		D	Input or preset nominal parameter adjustment size	/PARAM /(	357)	SDINP	O	DELP
						TOPM	D	DELP
DIP1		I	Phase initial times for nominal trajectory [sd]	/GENF /(	453)	GETIT	I	DIP1
						SDINP	M	DIP1
						TEST	O	DIP1
						TOPM	I	DIP1
DIS1		I	Arc initial times for nominal trajectory [sd]	/GENF /(	473)	GETIT	I	DIS1
						SDINP	M	DIS1
						TEST	O	DIS1
						TOPM	I	DIS1
						TRAN3	I	DIS1
DPAR	δp	D	Adjustable parameter corrections	/PARAM /(	264)	ADJUST	I	DPAR
						MTX3A	M	DPAR
						TOPM	D	DPAR
ID		I	A four word array containing the basic deck, reference run, case and part case numbers in that order.	/GLOBAL/(	21)	BLICO	I	ID
						FRENCH	I	ID
						GEINP	I	ID
						PADS1	D	ID
						PRINT	I	ID
						SDINP	I	ID
						TOPM	I	ID
						VEHDF	I	ID
IDPAY	$d\Phi$	D	Initial payoff improvement	/STS /(	1)	PAYO2	I	DPAY
						SDINP	I	DPAY
						SDINP	O	IST
						SDINP	I	ST
						TEST	I	DPAY
						TOPM	D	IDPAY
IFAR		M	File where nominal trajectory data is read from.	/XCODES/(	139)	AGETB3	I	IFAR
						AST3	I	IFAR
						TOPM	M	IFAR
IFATAL		M	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADS1	I	IFATAL
						SDINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
IFAW		M	Logical file to write current trial trajectory data	/XCODES/(	138)	AST3	I	IFAW
						TOPM	M	IFAW
IFB		O	File where adjoint solution is stored	/XCODES/(	140)	BGET3	I	IFB
						BSTO3	I	IFB
						TOPM	O	IFB
IFOB		O	Forward or adjoint integration flag = 1 means forward = 2 means adjoint	/XCODES/(	178)	ACCEL	I	IFOB
						BEROCO	I	IFOB
						BLYNE	I	IFOB
						EQUA3	I	IFOB
						IMPUL	I	IFOB
						SPLYNE	I	IFOB
						TOPM	O	IFOB
IITQ		D	Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
						ADIC3A	I	ITQ
						ADID3A	I	ITQ
						CON3	I	ITQ
						SDINP	M	ITQ
						STAU	I	ITQ
						TOPM	D	IITQ

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IOMG	Ω_J	D	Array of arc cut off values [sd]	/GENF /(	1)	ADJUST	M	OMG
						FNTG	I	OMG
						PRMSET	M	OMG
						PROPB	I	OMG
						SDINP	M	OMG
						STP3	I	OMG
						TOPM	D	IOMG
IPOINT		D	Code for each adjustable parameter in steepest descent.	/PARAM /(	1)	ADJUST	I	IPOINT
						PRMSET	I	IPOINT
						SDINP	D	IPOINT
						STAU	I	IPOINT
						TOPM	D	IPOINT
IPRINT		D	Print page counter initialization flag	/XCODES/(	168)	OUT	M	IPRINT
						TEST	I	IPRINT
						TOPM	D	IPRINT
ISTART		M	Initialization and divergance flag	/XCODES/(	147)	AST3	D	ISTART
						BLGCON	D	ISTART
						BLYNE	D	ISTART
						FNTG	I	ISTART
						MODELA	D	ISTART
						PROPIN	D	ISTART
						REU3	I	ISTART
						TEST	M	ISTART
						TOPM	M	ISTART
ITCT		M	Iteration counter	/XCODES/(	148)	BNTG	I	ITCT
						OUT	I	ITCT
						TEST	M	ITCT
						TOPM	M	ITCT
ITER		M	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
						FNTG	I	ITER
						GETIT	I	ITER
						MODELA	I	ITER
						OUT	I	ITER
						PAYO2	M	ITER
						PROPIN	I	ITER
						TEST	M	ITER
						TOPM	M	ITER
I2OP		D	First optimization pass flag sets dΦ = DPAY Also used to indicate payoff degradation due to restoration of constraints	/XCODES/(	136)	PAYO2	M	I2OP
						TEST	M	I2OP
						TOPM	D	I2OP
KOP		I	Counts number of times constraint misses are halved down because of divergance problems	/XCODES/(	154)	TEST	M	KOP
						TOPM	I	KOP
KWOW	v	D	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	D	VAR
						ADJUST	M	VAR
						AGETB3	D	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	D	KWOW
						VREF3	M	V

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
LUM		M	Program control flag. LUM = 0 Steepest descent only; LUM = 1 Steepest descent and adjoint transformation stored on tape; LUM = 2 Steepest descent and QL, LUM = 3. QL only.	/GLOBAL/(	6)	AST3	I	LUM
						FNTG	I	LUM
						GEINP	I	LUM
						PAOS1	M	LUM
						SDINP	I	LUM
						TOPM	M	LUM
MIXA		D	Maximum number of words in trajectory data buffer = 990	/RETREV/(	12)	AGETB3	I	MIXA
						AST3	I	MIXA
						SDINP	I	MIXA
						TOPM	D	MIXA
MIXB		D	Maximum number of words in adjoint data buffer = 3000	/RETREV/(	13)	BGET3	I	MIXB
						BST03	I	MIXB
						SDINP	I	MIXB
						TOPM	D	MIXB
NBFA		D	Maximum number of buffers permitted to store forward trajectory data = 20.	/RETREV/(	10)	AST3	I	NBFA
						TOPM	D	NBFA
NBFB		D	Maximum number of buffers permitted to store adjoint solution data = 60	/RETREV/(	11)	BST03	I	NBFB
						TOPM	D	NBFB
NCASE		O	Case number	/XCODES/(	159)	TOPM	O	NCASE
NCN		I	Number of elements in dΨ	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BST03	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAY02	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRIOSZ	I	NCN
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BST03	I	NCNST
						CON3	I	NCNST
						PAY02	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BST03	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
NEQF		O	Number of equations to be integrated on forward trajectory	/XCODES/(	185)	REU3	I	NEQF
						RKTA3A	I	NN
						SDINP	O	NEQF
						STAU	I	NEQF
						TOPM	O	NEQF
						TRAN3	O	NEQF
NOP		I	Counts number of times payoff is scaled down due to divergance problems	/XCODES/(	163)	TEST	M	NOP
						TOPM	I	NOP

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NPA		O	Running count of number of adjustable parameters to be perturbed on remainder of trajectory	/PARAM /(	14)	ADJUST	M	NPAR
						FNTG	I	NPA
						MTX3A	I	NPA
						TOPM	O	NPA
NPARA		O	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	O	NPARA
NPH		I	Number of phases in trajectory	/XCODES/(	164)	BNTG	I	NPH
						FNTG	O	NPH
						PRMSET	I	NPH
						SDINP	M	NPH
						TEST	I	NPH
						TOPM	I	NPH
NST		I	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST
OMEGA	ω	O	Earth rotation rate (RAD/SEC)	/STATE3/(	719)	BL4	I	OMEGA
						BL7	I	OMEGA
						TOPM	O	OMEGA
OMEGA2	ω^2	O	See symbol	/STATE3/(	720)	BL4	I	OMEGA2
						BL7	I	OMEGA2
						BL8	I	OMEGA2
						TOPM	O	OMEGA2
OMGZ	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
						CRASH	I	OMEGA
						DER3A	I	OMGZ
						EQUA3	I	OMGZ
						GEINP	I	OMGZ
						MODELA	I	OMGZ
						MODELB	I	OMGZ
						PDBC	I	OMGZ
						PDY3A	I	OMGZ
						SDINP	I	OMGZ
						TOPM	I	OMGZ
PARA	p	O	Adjustable parameter nominal values.	/PARAM /(	252)	ADJUST	I	PARA
						PRMSET	M	PARA
						TOPM	O	PARA
SPARA	$S^{\Psi}{}_{I}$	O	Matrix of adjustable parameter sensitivities (including all parameters)	/PARAM /(	15)	ADJUST	I	SPARA
						PAYO2	I	SPARA
						STAU	M	SPARA
						TOPM	O	SPARA
SPARB	$S^{\Psi}{}_{I}$	O	Matrix of adjustable parameter sensitivities (Contains only elements corresponding to parameters yet to be adjusted)	/PARAM /(	144)	ADJUST	O	SPARB
						MTX3A	I	SPARB
						TOPM	O	SPARB
SVAR	$y\|_{t=0}$	I	Array of state values at initial problem time [sd]	/GENF /(	79)	ADJUST	O	SVAR
						BNTG	I	SVAR
						FNTG	I	SVAR
						PRMSET	M	SVAR
						REU3	I	SVAR
						SDINP	M	SVAR
						TEST	I	SVAR
						TOPM	I	SVAR
						TRTOSZ	I	SVAR

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
S2INV	[SS]	D	Parameter sensitivity contribution to A matrix	/PARAM /(	276)	ADJUST	O	S2INV
						MTX3A	M	S2INV
						PAYO2	M	S2INV
						TOPM	D	S2INV
TPH1		I	Phase end times for nominal trajectory	/GENF /(	413)	BNTG	I	TPH1
						GETIT	I	TPH1
						SDINP	O	TPH1
						TEST	O	TPH1
						TOPM	I	TPH1
TST1		I	Arc end times for nominal trajectory	/GENF /(	433)	BNTG	I	TST1
						GETIT	I	TST1
						PROFIN	I	TST1
						SDINP	O	TST1
						TEST	O	TST1
						TOPM	I	TST1
						TRAN3	I	TST1
						TRTDSZ	I	TST1
WTP	[Y]	D	Adjustable parameter diagonal weighting matrix order according to IPOINT.	/PARAM /(	132)	MTX3A	I	WTP
						PAYO2	I	WTP
						SDINP	O	WTP
						TOPM	D	WTP
WTPD		D	Input or preset adjustable parameter weighting factors according to type of parameter	/PARAM /(	123)	SDINP	M	WTPD
						TOPM	D	WTPD

30 OCT 72 G 01-46

```
      PROGRAM TOPM                                                       TOPM
C                                                                        COMN
C      STEEPEST  DESCENT  MAIN  PROGRAM                                  COMN
C                                                                        COMN
      COMMON/GENF/                                                       GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WOC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,COE       ,DELTAE    ,SIO      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                   GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED   ,IRATED      ,                                            GENF
     *P1         ,P2        ,P3      ,XK1         ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1Q      ,XK2Q      ,XK3Q     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DFDY(3,8)  GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                        GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                 GENF
      COMMON/STS/                                                        STS
     *DPAY       ,PMIN      ,WORK (20),NWDS      ,IPC    (7),NITER     , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)            STS
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                            RETREV
      COMMON/RETREV/                                                     RETREV
     *NBUFA(2)   ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA     , RETREV
     *MIXB       ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB      RETREV
     *,NBUFB     ,IBUFB                                                  RETREV
      COMMON /XCODES/                                                    XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JSID(20,2),JPH (20,2),XCODES
     *JST    (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD.      ,NCASE    , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JSII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/DATA/                                                       DATA
     *PI         ,RAD       ,RDI       ,SC        ,UMF       ,TMPF     , DATA
     *FTNM       ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4       DATA
      COMMON/GLOBAL/                                                     GLOBAL
     *GR         ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM        GLOBAL
     *,JJOP(10)  ,IFATAL    ,NARC        ,NBRAN     ,NFARC     ,IO(4)    GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG     ,MAXTAB                              GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)      GLOBAL
     *,ITPSO  ,KSOL      ,KGLOBL(8)                                      RETAP
      COMMON/ ORBIT/  VI,       GAMI,      PSII,      XMUI,        P,    ORBIT
     *       ECC,      AINCL,      ARGP,    ASCNOD,    SMIMAJ,   APOGEE, ORBIT
     *  PERGEE,    ANOMLY,        CAPX,       CAPY,     ASYMP,   ENERGY, ORBIT
     *    HMNTM,     DVIDV,     DVIDG,                                   ORBIT
     *  DVIDH ,      DVIDM,    DVIDPS,    DVIDRO,    DVIDMU,     DGIDV,  ORBIT
     *  DGIDG ,      DGIDH,    DGIDM ,    DGIDPS,    DGIDRO,     DGIDMU, ORBIT
     *  DPIDV ,      DPIDG,    DPIDH ,    DPIDM ,    DPIDPS,     DPIDRO, ORBIT
     *  DPIDMU,      DMIDV,    DMIDG ,    DMIDH ,    DMIDM ,     DMIDPS, ORBIT
     *  DMIDRO,      DMIDMU,   DPDV  ,    DPDG  ,    DPDH  ,     DPDM  , ORBIT
     *  DPDPS ,      DPDRO ,   DPDMU ,    DECDV ,    DECDG ,     DECDH , ORBIT
```

20 OCT 72 G 01-46

```
      * DECDM ,     DECDPS, DECDRO,   DECDMU,   DIDV  ,    DIDG  ,          ORBIT
      * DIDH  ,     DIDM  , DIDPS ,   DIDRO ,   DIDMU ,    DBEDV ,          ORBIT
      * DBEDG ,     DBEDH , DBEDM ,   DBEDPS,   DBEDRO,    DBEDMU,          ORBIT
      * DNODV ,     DNODG , DNODH ,   DNODM ,   DNODPS,    DNODRO,          ORBIT
      * DNODMU,     DSMDV,   DSMDG,    DSMDH,    DSMDM,    DSMDPS           ORBIT
       COMMON/ORBIT/                                                        ORBIT
      * DSMDRO,   DSMDMU,    DAPDV,    DAPDG,    DAPDH,    DAPDM ,          ORBIT
      * DAPDPS,   DAPDRO,    DAPDMU,   DPEDV,    DPEDG,    DPEDH ,          ORBIT
      *DPEDM     ,DPEDPS    ,DPEDRO    ,DPEDMU    ,DANDV     ,DANDG     ,   ORBIT
      * DANDH ,   DANDM ,    DANDPS,   DANDRO,   DANDMU,   DCXDV ,          ORBIT
      * DCXDG ,   DCXDH ,    DCXDM ,   DCXDPS,   DCXDRO,   DCXDMU,          ORBIT
      * DCYDV ,   DCYDG ,    DCYDH ,   DCYDM ,   DCYDPS,   DCYDRO,          ORBIT
      * DCYDMU,   DASDV ,    DASDG ,   DASDH ,   DASDM ,   DASDPS,          ORBIT
      * DASDRO,   DASDMU,    DENDV ,   DENDG ,   DENDH ,   DENDM ,          ORBIT
      * DENDPS,   DENDRO,    DENDMU,   DMODV ,   DMODG ,   DMODH ,          ORBIT
      * DMODM ,   DMODPS,    DMODRO,   DMODMU                               ORBIT
       DIMENSION ORBPRM(18),PPO(7,18)                                       ORBIT
       EQUIVALENCE (VI,ORBPRM),(DVIDV,PPO)                                  ORBIT
       COMMON/ORBIT/ YAXRF,SNXLMF,CSXLMF,SDOWN,SCROSS,TD,TC                 ORBIT
      *,SNPSR     ,CSPSR     ,SNGI      ,CSGI      ,SPSII     ,CPSII     ,  ORBIT
      * STOT      ,CSI       ,SNI       ,SNGNU     ,CSAND     ,COSDMU    ,  ORBIT
      * SINDMU    ,THT       ,WTFUEL                                        ORBIT
       COMMON/STATE3/                                                       STATE3D
      *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,Y0(9)     ,SVY(10)   ,  STATE3D
      *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) ,  STATE3D
      *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,DCORHO    ,DCORO2    ,  STATE3D
      *SVBV   (9),OMEGA     ,OMEGA2    ,                                    STATE3D
      *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       ,  STATE3D
      *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       ,  STATE3D
      *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       ,  STATE3D
      *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       ,  STATE3D
      *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       ,  STATE3D
      *PDO        ,UDO       ,HTDV      ,HTDR                               STATE3D
       REAL MDM , MDV, MDR                                                  STATE3D
       COMMON/STATE3/                                                       STATE3D
      *SIN2RO     ,COS2RO    ,COS2GM                                        STATE3D
       DIMENSION IITO(219)  ,IOMG(633),    IOPAY(370)                       PH1SZ
       EQUIVALENCE (ITO,IITO),(OMG,IOMG),(OPAY,IOPAY)                       TOPM
       COMMON/PARAM/                                                        PARAM
      *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),              PARAM
      *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                          PARAM
      *,DELP(9)                                                             PARAM
       DATA  S2INV / 81* 0. /                                               TOPM
       DATA IPOINT,NPARA,NPA,SPARA,WTP,SPARB,PARA,DPAR/                     PH1SZ
      * 12*0,0,0,108*0.,12*0.,108*0.,12*0.,12*0./                           PH1SZ
       LOGICAL IFATAL                                                       TOPM
       DATA WIPD/1.,1.,1.,.01,1.,2.,1.,.1,0./                               TOPM
       DATA DELP / .02,2.,02,200.,2.,.02,.01,.01, 0./                       TOPM
       DATA IOMG/633*0/                                                     PH1SZ
       DATA IITO/219*0/                                                     PH1SZ
       DATA IOPAY/370*0/                                                    PH1SZ
       DATA MIXA,MIXB,NBFA,NBFB / 990,3000,20,60 /                          TOPM
 C                                                                          TOPM
       DIMENSION KWOW(757)                                                  PH1SZ
       EQUIVALENCE(KWOW,VAR)                                                PH1SZ
       DATA KWOW / 757*0/                                                   TOPM
    10 CONTINUE                                                             TOPM
    20 CONTINUE                                                             TOPM
 C       I     SCAN  INPUT  DATA                                            COMN
       CALL OVERLAY(4HSDIN,3,1,0)                                           TOPM
 C       II    IF  INPUT  ERRORS  RETURN                                    COMN
       IF(IFATAL) GO TO 130                                                 TOPM    130
 C       III  IF  OL  ONLY   RETURN                                         COMN
       IF(LUM.EQ.3) GO TO 130                                               TOPM    130
 C        SET  FILE  NUMBERS  AND  FLAG  FOR  FIRST  NOMINAL                COMN
       IFAW =39                                                             TOPM
       IFAR =40                                                             TOPM
       IFB  =41                                                             TOPM
       ISTART =1                                                            TOPM
       I2OP =0                                                              TOPM
       ITER =1                                                              TOPM
       IPRINT =1                                                            TOPM
       ITCT=1                                                               TOPM
       NCASE= ID(3)                                                         TOPM
       OMEGA=OMGZ                                                           TOPM
```

20 OCT 72 6.01-96

```
      OMEGA2=OMGZ*OMGZ                                                  TOPM
   30 CONTINUE                                                          TOPM
      IFOB =1                                                           TOPM
C                                                                       COMN
C       IV      INTEGRATE   TRAJECTORY                                  COMN
C       V       IF  NO CONTROL DIVERGANCE  ERROR  GO  TO  VIII          COMN
C               ELSE   CHECK  IF  FIRST NOMINAL   GO TO  XVIII          COMN
C               ELSE GO TO VI                                           COMN
      CALL OVERLAY (5HIFNTG,3,2,6HRECALL)                               TOPM
      IF(ISTART.NE.6) GO TO 40                                          TOPM
      IF(ITCT.LE.1) GO TO 125                                           PHISZ
C       VI    CALL TEST TO SCALE DOWN STEP SIZE(S)                      COMN
      CALL OVERLAY ( 5HITEST,3,3,0)                                     TOPM
      IPRINT =1                                                         TOPM
C       VII   TEST   WHETHER  SOLUTION  CONVERGANCE HAS FALTERED        COMN
C         IF  SO  GO  TO  XI    ELSE  RESET FLAG AND GO TO IX           COMN
      IF(ISTART.EQ.5) GO TO 70                                          TOPM
      ISTART =2                                                         TOPM
      GO TO  60                                                         TOPM
C       VIII    NORMAL  SOLUTION  CONVERGANCE  TESTING                  COMN
C                                                                       COMN
C       VIII-A    IF  SOLUTION  TRAJECTORY HAS  ALREADY BEEN  INTGRATED COMN
C                 RETURN  ELSE  GO TO  VIII-B                           COMN
   40 CONTINUE                                                          TOPM
      IF(ITER.EQ.3) GO TO 130                                           TOPM
      IPRINT=1                                                          TOPM
C       VIII-B      TEST   CONVERGANCE                                  COMN
   50 CALL  OVERLAY(5HITEST,3,3,0)                                      TOPM
   60 CONTINUE                                                          TOPM
C                                                                       COMN
C       IX   PRINT  CONSTRAINT  MISSES AND  PAYOFF                      COMN
      CALL IPR(4HDCON,DCON,I,NCN,0)                                     TOPM
      IF(ITER.EQ.2) CALL IPR(6HPAYOFF,BCON(NCN),I,1,0)                  TOPM
      IF(ISTART EQ.1) ISTART=2                                          TOPM
C        X   IF  NO  CONVERGANCE  PROBLEM  GO  TO  XII  ELSE  XI        COMN
      IF(ISTART.NE.5) GO TO 80                                          TOPM
C        XI    SET   SOLUTION  FLAG   GO  TO  XVII                      COMN
   70 ITER=3                                                            TOPM
      ISTART=2                                                          TOPM
      GO TO 110                                                         TOPM
C                                                                       COMN
C         XII    IF  SOLUTION FLAG NOT  SET  GO  TO  XIII               COMN
C                ELSE   REVERSE  READ AND WRITE FILES AND GO TO XVII    COMN
   80 CONTINUE                                                          TOPM
      IF(ITER.NE.3) GO TO 90                                            TOPM
      IFR=IFAR                                                          TOPM
      IFAR=IFAW                                                         TOPM
      IFAW=IFR                                                          TOPM
      GO TO 110                                                         TOPM
C                                                                       COMN
C         XIII    IF STEP-SIZE  SCALING HAS  OCCURRED  RUN  ANOTHER     COMN
C                 TRIAL  TRAJECTORY ( GO TO  IV )                       COMN
C                 ELSE  RESET  FILE NUMBERS  GO  TO  XIV                COMN
   90 IF(KOP+NOP.GT.0) GO TO 30                                         TOPM
      IFR = IFAR                                                        TOPM
      IFAR= IFAW                                                        TOPM
      IFAW = IFR                                                        TOPM
      IFOB =2                                                           TOPM
C                                                                       COMN
C         XIV      INTEGRATE  ADJOINT  SOLUTION                         COMN
      CALL  OVERLAY( 5HIBNTG,3,4,0)                                     TOPM
      ISTART=2                                                          PHISZ
C                                                                       COMN
C        XV        IF  OPTIMIZATION  PASS  COMPUTE PAYOFF IMPROVEMENT   COMN
      IF( ITER.NE.2) GO TO 100                                          TOPM
      CALL  OVERLAY (5HIPAYO,3,5,0)                                     TOPM
  100 CALL IPR( 4HDCON, DCON,I,NCN,0)                                   TOPM
      IF(ITER.EQ.3) GO TO 110                                           TOPM
C       XVI        IF  SOLUTION  FLAG SET  GO TO XVII  ELSE GO TO IV    COMN
      GO TO  30                                                         TOPM
  110 CONTINUE                                                          TOPM
```

```
C      XVII       IF  NO  TRANSFORMATION FLAG SET  GO TO .XVII-C               COMN
       IF(LUM EQ.0) GO TO 120                                                  TOPM
       IF(NCN EQ NCNST+1) GO TO 115                                            PHISZ
C       XVII-A    TRANS.FLAG IS SET, IF LAST TRAJECTORY WAS                    COMN
C                 OPTIMIZATION  PASS  GO TO  XVII-B                            COMN
C                 ELSE  ERROR  FLAG  SET  ,GO TO  XVII-C                       COMN
       IFATAL = .TRUE.                                                         PHISZ
       LUM=0                                                                   PHISZ
       GO TO 120                                                               PHISZ
   115 CONTINUE                                                                PHISZ
C      XVII-B       CALCULATE  TRANSFORMED  ADJOINTS                           COMN
C                   WRITE FIRST STARTING SOLUTION  RECORD  AND  GO TO IV       COMN
       ITER=2                                                                  TOPM
       CALL OVERLAY(4HTRAN,3,6,0)                                              TOPM
       REWIND 11                                                               PHISZ
       WRITE(11) SVAR(1),NEQ,NST,(DIS1(I),I=1,NST),(DIP1(J),J=1,NPH)           RETAP
      *,(     TST1(II),II=1,NST),(TPH1(IJ),IJ=1,NPH)                           NOS
       ITER=3                                                                  TOPM
       GO TO 30                                                                TOPM
C      XVII-C       RESET  NO. OF  EQUATIONS, GO TO IV                         COMN
   120 NEQF = NEQ +3                                                           TOPM
       GO TO30                                                                 TOPM
C      XVIII      FATAL  ERROR  RETURN                                         COMN
   125 IFATAL=.TRUE.                                                           PHISZ
   130 CONTINUE                                                                TOPM
       END                                                                     TOPM
```

20 OCT 72 G 01-46

BLØCK
AEC03

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
APHO	α_{old}	Angle of attack from last nominal trajectory	(DEG)	/AEC03 /(	1)	AST3	M	APHO
						FNTG	I	APHO
						MTX3A	I	APHO
						OUT	I	APHO
						PROPB	O	AEZRO
						PROPIN	O	AEZRO
APHR	α	Angle of attack	(DEG)	/AEC03 /(	2)	AGETB3	O	APHR
						AST3	M	APHR
						BEROCO	I	APHR
						BLGCON	O	APHR
						GUI3A	M	APHR
						MODELA	M	APHR
						MODELB	I	APHR
						MTX3A	O	APHR
						OUT	I	APHR
ALPHA	α	Angle of attack	(RAD)	/AEC03 /(	3)	BEROCO	I	ALPHA
						BLGCON	M	ALPHA
						BL2	I	ALPHA
						FNTG	O	ALPHA
						MAMECO	I	ALPHA
						MODELA	M	ALPHA
						MODELB	O	ALPHA
						REU3	O	ALPHA
						VT	I	ALPHA
VDA	$\partial\dot{v}/\partial\alpha$	See symbol		/AEC03 /(	4)	ACCEL	O	VDA
						ADEQ3A	I	VDA
GDA	$\partial\dot{\gamma}/\partial\alpha$	See symbol		/AEC03 /(	5)	ACCEL	O	GDA
						ADEQ3A	I	GDA
PDA	$\partial\dot{\psi}/\partial\alpha$	See symbol		/AEC03 /(	6)	ACCEL	O	PDA
						ADEQ3A	I	PDA
SINA	$\sin\alpha$	See symbol		/AEC03 /(	7)	ACCEL	I	SINA
						BL4	I	SINA
						BL6	I	SINA
						BL7	I	SINA
						BL8	I	SINA
						FH3	I	SINA
						GUI3A	M	SINA
						OUT	I	SINA
						VT	M	SINA
COSA	$\cos\alpha$	See symbol		/AEC03 /(	8)	ACCEL	I	COSA
						BL4	I	COSA
						BL6	I	COSA
						BL7	I	COSA
						BL8	I	COSA
						FH3	I	COSA
						OUT	I	COSA
						VT	M	COSA
PHIO	ϕ_{old}	Bank angle from last nominal trajectory		/AEC03 /(	9)	AST3	M	PHIO
						MTX3A	I	PHIO
PHID	ϕ	Bank angle	(DEG)	/AEC03 /(	10)	AGETB3	O	PHID
						AST3	M	PHID
						GUI3A	M	PHID
						MODELA	M	PHID
						MODELB	I	PHID
						MTX3A	O	PHID
						OUT	I	PHID
PHI	ϕ	Bank angle	(RAD)	/AEC03 /(	11)	GUI3A	M	PHIR
						MODELA	M	PHI
						MODELB	M	PHI
						OUT	I	PHI
SINPHI	$\sin\phi$	See symbol		/AEC03 /(	12)	ACCEL	I	SINPHI
						BL4	I	SINPHI
						MODELA	M	SINPHI
						MODELB	M	SINPHI
						OUT	I	SINPHI

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
COSPHI	$\cos\phi$	See symbol	/AEC03 /(	13)	ACCEL	I	COSPHI
					BL4	I	COSPHI
					GUI3A	M	COSPHI
					MODELA	M	COSPHI
					MODELB	M	COSPHI
					OUT	I	COSPHI
GDPHI	$\partial\dot{\gamma}/\partial\phi$	See symbol	/AEC03 /(	14)	ACCEL	O	GDPH
					ADEQ3A	I	GDPH
					PDY3A	O	GDPH
PDPHI	$\partial\dot{\psi}/\partial\phi$	See symbol	/AEC03 /(	15)	ACCEL	O	PDPH
					ADEQ3A	I	PDPH
					PDY3A	O	PDPH
XLAMA	$\Lambda^{\psi_i \Omega_j}$	Impulse response function column vector associated with angle of attack	/AEC03 /(	16)	ADEQ3A	M	XLAMA
					ADIC3A	O	XLAMA
					AST3	O	XLAMA
					BGET3	O	XLAMA
					BST03	M	XLAMA
					MTX3A	I	XLAMA
					TRAN3	M	XLAMA
XLAMP	$\Lambda^{\psi_i \Omega_j}$	Impulse response function column vector associated with bank angle	/AEC03 /(	25)	ADEQ3A	M	XLAMP
					ADIC3A	O	XLAMP
					AST3	O	XLAMP
					BGET3	O	XLAMP
					BST03	M	XLAMP
					MTX3A	I	XLAMP
					TRAN3	M	XLAMP
CD0	C_{D_0}	Drag coefficient at $\alpha = 0$	/AEC03 /(	34)	BEROCO	I	CD0
					EQUA3	I	CD0
CD0M	$\partial C_{D_0}/\partial M$	See symbol	/AEC03 /(	35)	BEROCO	I	CD0M
					EQUA3	I	CD0M
CL0	C_{L_0}	Lift coefficient at $\alpha = 0$	/AEC03 /(	36)	BEROCO	I	CL0
					EQUA3	I	CL0
FK	k	Induced drag coefficient	/AEC03 /(	37)	BEROCO	I	FK
					EQUA3	I	FK
XCGM	$\partial X_{CG}/\partial m$	See symbol	/AEC03 /(	38)	EL2	I	XCGM
					EQUA3	O	XCGM
					VT	I	XCGM
ZCGM	$\partial Z_{CG}/\partial m$	See symbol	/AEC03 /(	39)	EL2	I	ZCGM
					EQUA3	O	ZCGM
					VT	I	ZCGM
CL0M	$\partial C_{L_0}/\partial M$	See symbol	/AEC03 /(	40)	BEROCO	I	CL0M
					EQUA3	I	CL0M
CM	C_M	Moment coefficient	/AEC03 /(	41)	MAMECO	O	CM
					VT	I	CM
CMA	C_{M_α}	Moment coefficient slope	/AEC03 /(	42)	EQUA3	M	CMA
					MAMECO	I	CMA
					VT	I	CMA
CMAM	$\partial C_{M_\alpha}/\partial M$	See symbol	/AEC03 /(	43)	EQUA3	M	CMAM
					MAMECO	I	CMAM
CMM	$\partial C_M/\partial M$	See symbol	/AEC03 /(	44)	MAMECO	O	CMM
					VT	I	CMM
CM0	C_{M_0}	Moment coefficient at $\alpha = 0$	/AEC03 /(	45)	EQUA3	I	CM0
					MAMECO	I	CM0
CM0M	$\partial C_{M_0}/\partial M$	See symbol	/AEC03 /(	46)	EQUA3	I	CM0M
					MAMECO	I	CM0M
FKM	$\partial k/\partial M$	See symbol	/AEC03 /(	47)	BEROCO	I	FKM
					EQUA3	I	FKM
CLAM	$\partial C_{L_\alpha}/\partial M$	See symbol	/AEC03 /(	48)	BEROCO	I	CLAM
					EQUA3	M	CLAM
CL	C_L	Lift coefficient	/AEC03 /(	49)	BEROCO	M	CL
					OUT	I	CL
					VT	I	CL

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CLA	C_{L_α}	Lift coefficient slope	/AEC03 /(	50)	BEROCO	M	CLA
					EQUA3	M	CLA
					VT	I	CLA
CLM	$\partial C_L/\partial M$	See symbol	/AEC03 /(	51)	BEROCO	M	CLM
					VT	I	CLM
CD	C_D	Drag coefficient	/AEC03 /(	52)	BEROCO	O	CD
					OUT	I	CD
					VT	I	CD
CDA	$\partial C_D/\partial\alpha$	See symbol	/AEC03 /(	53)	BEROCO	M	CDA
					VT	I	CDA
CDM	$\partial C_D/\partial M$	See symbol	/AEC03 /(	54)	BEROCO	O	CDM
					VT	I	CDM

BLØCK
AIRBRE

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
TAIRB		Airbreather thrust.	(LBS)	/AIRBRE/(	1)	EQUA3 FH4	I I	TAIRB TAIRB
TAIRBV		Partial of airbreather thrust WRT velocity		/AIRBRE/(	2)	ACCEL FH4	I I	TAIRBV TAIRBV
TAIRBH		Partial of airbreather thrust WRT altitude		/AIRBRE/(	3)	ACCEL FH4	I I	TAIRBH TAIRBH
SFC		Specific fuel consumption	(LBS/LB/HR)	/AIRBRE/(	4)	ACCEL	I	SFC
SFCV		Partial of SFC WRT velocity.		/AIRBRE/(	5)	ACCEL	I	SFCV
SFCH		Partial of SFC WRT altitude.		/AIRBRE/(	6)	ACCEL	I	SFCH

BLØCK
AXL

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AV	a^{v}	Acceleration vector element.	/AXL	/(1)	ACCEL DER3A	M I	AV AV
AG	a^{γ}	Acceleration vector element.	/AXL	/(2)	ACCEL DER3A PDY3A	M I I	AG AG AG
AP	a^{ψ}	Acceleration vector element.	/AXL	/(3)	ACCEL DER3A PDY3A	M I I	AP AP AP
AM	a^{m}	Acceleration vector element.	/AXL	/(4)	ACCEL DER3A	M I	AM AM
AY		Name of acceleration partials matrix.	/AXL	/(5)	ACCEL ACCEL PDY3A	M O I	AVV AY AVV
AVV		Element of matrix of acceleration vector partials WRT state.	/AXL	/(5)			
AGV		Element of matrix of acceleration vector partials WRT state.	/AXL	/(6)	ACCEL PDY3A	O I	AGV AGV
APV		Element of matrix of acceleration vector partials WRT state.	/AXL	/(7)	ACCEL PDY3A	O I	APV APV
AMV		Element of matrix of acceleration vector partials WRT state.	/AXL	/(8)	ACCEL PDY3A	O I	AMV AMV
AVG		Element of matrix of acceleration vector partials WRT state.	/AXL	/(9)			
AGG		Element of matrix of acceleration vector partials WRT state	/AXL	/(10)			
APG		Element of matrix of acceleration vector partials WRT state.	/AXL	/(11)			
AMG		Element of matrix of acceleration vector partials WRT state.	/AXL	/(12)			
AVP		Element of matrix of acceleration vector partials WRT state.	/AXL	/(13)			
AGP		Element of matrix of acceleration vector partials WRT state.	/AXL	/(14)			
APP		Element of matrix of acceleration vector partials WRT state.	/AXL	/(15)			
AMP		Element of matrix of acceleration vector partials WRT state.	/AXL	/(16)			
AVR		Element of matrix of acceleration vector partials WRT state.	/AXL	/(17)	ACCEL PDY3A	O I	AVR AVR
AGR		Element of matrix of acceleration vector partials WRT state	/AXL	/(18)	ACCEL PDY3A	O I	AGR AGR
APR		Element of matrix of acceleration vector partials WRT state.	/AXL	/(19)	ACCEL PDY3A	O I	APR APR
AMR		Element of matrix of acceleration vector partials WRT state.	/AXL	/(20)	ACCEL PDY3A	O I	AMR AMR
AVO		Element of matrix of acceleration vector partials WRT state.	/AXL	/(21)			
AGO		Element of matrix of acceleration vector partials WRT state.	/AXL	/(22)			
APO		Element of matrix of acceleration vector partials WRT state	/AXL	/(23)			
AMO		Element of matrix of acceleration vector partials WRT state	/AXL	/(24)			
AVU		Element of matrix of acceleration vector partials WRT state.	/AXL	/(25)			
AGU		Element of matrix of acceleration vector partials WRT state.	/AXL	/(26)			
APU		Element of matrix of acceleration vector partials WRT state.	/AXL	/(27)			

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AMU		Element of matrix of acceleration vector partials WRT state	/AXL	/(28)			
AVM		Element of matrix of acceleration vector partials WRT state.	/AXL	/(29)	ACCEL PDY3A	O I	AVM AVM
AGM		Element of matrix of acceleration vector partials WRT state.	/AXL	/(30)	ACCEL PDY3A	O I	AGM AGM
APM		Element of matrix of acceleration vector partials WRT state.	/AXL	/(31)	ACCEL PDY3A	O I	APM APM
AMM		Element of matrix of acceleration vector partials WRT state.	/AXL	/(.32)	PDY3A	I	AMM

BLØCK DATA

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
PI	π	Constant 3.141592653	/DATA /(	1)	OUT	I	PI
					PADS1	D	PI
RAD		Radian to angle conversion, 57 29577951	/DATA /(	2)	BEROCO	I	DEG
					BLGCON	I	RAD
					ENVPRM	I	RAD
					EDUA3	I	RAD
					FNTG	I	RAD
					GUI3A	I	RAD
					MODELA	I	RAD
					MTX3A	I	RAD
					OUT	I	RAD
					PADS1	D	RAD
					SDINP	I	RAD
					TRTOSZ	I	RAD
RDI		Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO	I	RDI
					DER3A	I	RDI
					FNTG	I	RDI
					GUI3A	I	RDI
					MODELA	I	RDI
					MODELB	I	RDI
					PADS1	D	RDI
					PROPB	I	RDI
					PROPIN	I	RDI
					REU3	I	RDI
					SDINP	I	RDI
					SOMG	I	RDI
SC		Constant in Sutherlands equation, 198	/DATA /(	4)	PADS1	D	SC
					PAT63	I	SC
UMF		Constant in Sutherlands equation, $.301 \times 10^{-6}$	/DATA /(	5)	PADS1	D	UMF
					PAT63	I	UMUF
TMPF		Constant in Sutherlands equation, 392	/DATA /(	6)	PADS1	D	TMPF
					PAT63	I	TF
FTNM		Feet to naut. mi. conversion, $1.645791629 \times 10^{-4}$	/DATA /(	7)	OUT	I	FTNM
					PADS1	D	FTNM
					TRTOSZ	I	FTNM
CAR		Constant, 1715.4827	/DATA /(	8)	PADS1	D	CAR

BLØCK
GENF

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
OMG	Ω_J	Array of arc cut off values [sd]	/GENF	/(1)	ADJUST	M	OMG
					FNTG	I	OMG
					PRMSET	M	OMG
					PROPB	I	OMG
					SDINP	M	OMG
					STP3	I	OMG
					TOPM	O	IOMG
OMGP		Array of phase cut off values [sd]	/GENF	/(21)	ADJUST	O	OMGP
					FNTG	M	OMGP
					PRMSET	O	OMGP
					SDINP	M	OMGP
VARQ	(VARQ)	Desired constraint values [sd]	/GENF	/(61)	CON3	I	VARQ
					SDINP	M	VARQ
					TEST	I	VARQ
TOL		Tolerence on constraint misses [sd]	/GENF	/(70)	SDINP	M	TOL
					SUMS	I	TOL
					TEST	I	TOL
SVAR	$y\|_{t=0}$	Array of state values at initial problem time [sd]	/GENF	/(79)	ADJUST	O	SVAR
					BNTG	I	SVAR
					FNTG	I	SVAR
					PRMSET	M	SVAR
					REU3	I	SVAR
					SDINP	M	SVAR
					TEST	I	SVAR
					TOPM	I	SVAR
					TRTOSZ	I	SVAR
WDC		Array of drop weight per arc[sd] (LBS)	/GENF	/(89)	BNTG	I	WDC
					REU3	I	WDC
					SDINP	M	WDC
A	A	Control integral matrix	/GENF	/(109)	ADEQ3A	O	A
					ADICB3	M	A
					BGET3	O	A
					BNTG	I	A
					BSTO3	I	A
					MTX3A	I	A
					PAYO2	I	A
					SDINP	I	A
					TRAN3	I	A
ACON		Vector of nominal constraint misses + PAYOFF IMPROVEMENT	/GENF	/(190)	CON3	M	ACON
					TEST	M	ACON
BCON		Vector of constraint misses on trial trajectory	/GENF	/(199)	CON3	O	BCON
					TEST	I	BCON
					TOPM	I	BCON
COTI		Temp storage for a matrix also called B matrix	/GENF	/(208)	ADICB3	M	COTI
					MTX3A	M	B
					TRAN3	M	COTI
DCON	$d\Psi_i$	Asked for correction in constraint misses and payoff vector	/GENF	/(289)	CON3	O	DCON
					MTX3A	I	DCON
					PAYO2	M	DCON
					TEST	M	DCON
					TOPM	I	DCON
					TRTOSZ	I	DCON
DTP		Altered integration interval reqd to hit phase cut off [sd] (SEC)	/GENF	/(298)	FNTG	M	DTP
DTS		Altered integration interval reqd to hit arc cut-off [sd] (SEC)	/GENF	/(299)	FNTG	M	DTS
DT		Integration interval (SEC)	/GENF	/(300)	BNTG	M	DT
					FNTG	M	DT
					REU3	I	DT
					RKTA3A	I	P
					RKTB3A	I	P
					STP3	I	DT
					YREF3	O	DT

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
G	g	Gravitational attraction (FT/SEC**2)	/GENF	/(301)	BL4	I	G
					BL7	I	G
					BL8	I	G
					DER3A	I	G
					EQUA3	M	G
					MODELA	I	G
					MODELB	I	G
					PDY3A	I	G
					SDER3	I	G
					SDINP	M	G
DPSQ	$(dP)^2$	Metric of control and parameter changes [sd]	/GENF	/(302)	PAYO2	M	DPSQ
					TEST	M	PSISQ
					TRTOSZ	I	DPSQ
Q	q	Dynamic pressure (PSF)	/GENF	/(303)	ENVPRM	I	Q
					EQUA3	M	Q
					OUT	I	Q
					PDBC	I	Q
					VT	I	Q
QS		Product of dynamic pressure and aero. Ref. Area (LBS)	/GENF	/(304)	EQUA3	O	QS
					VT	I	QS
R	R	Radial distance from earth center to vehicle (FT)	/GENF	/(305)	BL4	I	R
					BL7	I	R
					BL8	I	R
					DER3A	I	R
					EQUA3	M	R
					MODELA	I	R
					MODELB	I	R
					PDBC	I	R
					PDY3A	I	R
					TRTOSZ	I	R
RE	R_{ey}	Unit reynolds number (1/FT)	/GENF	/(306)	OUT	M	RE
					PDBC	M	RE
MACH	M	Mach number	/GENF	/(307)	BEROCO	I	MACH
					ENVPRM	I	MACH
					EQUA3	M	MACH
					OUT	I	MACH
PA	P_a	Atmospheric pressure (PSF)	/GENF	/(308)	EQUA3	M	DZM
					FH2	I	PA
					IMPUL	I	PA
					OUT	I	PA
					PDBC	I	PA
					SDER3	I	PA
RO	ρ_a	Atmospheric density (SLUGS/FT**3)	/GENF	/(309)	BL7	I	RO
					BL8	I	RO
					DER3A	I	RO
					EQUA3	I	RO
					OUT	I	RO
					PDBC	I	RO
					PDY3A	I	RO
CS	a	Speed of sound (FT/SEC)	/GENF	/(310)	EQUA3	M	CS
					OUT	I	CS
VNU	μ_a	Atmospheric viscosity [dynamic] (SLUGS/FT/SEC)	/GENF	/(311)	OUT	I	VNU
					PDBC	I	VNU
PAR		Deriv. Of press. Wrt alt.	/GENF	/(312)	ACCEL	I	PAR
					FH2	I	PAR
ROR		Deriv. Of density wrt alt.	/GENF	/(313)	BL7	I	ROR
					BL8	I	ROR
					EQUA3	I	ROR
					PDBC	I	ROR
					PDY3A	I	ROR
CSR		Deriv Of speed of sound wrt alt.	/GENF	/(314)	EQUA3	I	CSR
VNR		Deriv.of viscosity wrt alt.	/GENF	/(315)	PDBC	I	VNR
SUMSQ		Sum of squares of constraint misses (BCON) divided by tolerances on trial trajectory	/GENF	/(316)	TEST	M	SUMSQ
SVSQ		Same as sumsq but saved for nominal trajectory [sd]	/GENF	/(317)	TEST	M	SVSQ

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
TIMEPH	τ_p	Phase time	(SEC)	/GENF	/(318)	EQUA3	O	TIMEPH
						FNTG	M	TIMEPH
						GETIT	I	TIMEPH
						GUI3A	I	TIMEPH
						OUT	I	TIMEPH
TIMES	τ	Arc time	(SEC)	/GENF	/(319)	AST3	I	TIMES
						EQUA3	O	TIMES
						FNTG	M	TIMES
						GETIT	I	TIMES
						OUT	I	TIMES
TOP		Elasped time at phase initiation		/GENF	/(320)	BNTG	M	TOP
						EQUA3	I	TOP
						FNTG	M	TOP
TOS		Elasped time at arc initiation		/GENF	/(321)	BNTG	M	TOS
						EQUA3	I	TOS
						FNTG	M	TOS
TR		Vector modifier of impulse response function in control calculation		/GENF	/(322)	MTX3A	M	TR
						TRAN3	M	TR
TST		Array of arc end times on trial trajectory [sd]		/GENF	/(331)	ADICB3	I	TST
						BNTG	I	TST
						FNTG	O	TST
						TEST	I	TST
TPH		Array of phase end times on trial trajectory [sd]		/GENF	/(351)	FNTG	O	TPH
						TEST	I	TPH
DIS		Array of arc end integration intervals for trial trajectory		/GENF	/(371)	BNTG	I	DIS
						FNTG	O	DIS
DIP		Array of phase end integration intervals for trial trajectory		/GENF	/(391)	BNTG	I	DIP
						FNTG	O	DIP
T	T	Thrust	(LBS)	/GENF	/(411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SDER3	I	T
W	W	Weight	(LBS)	/GENF	/(412)	BL5	I	W
						ENVPRM	I	W
						EQUA3	M	W
						FH3	I	W
						OUT	I	W
						PDBC	I	W
						REU3	I	W
						TRTOSZ	I	W
TLP1		Phase end times for nominal trajectory		/GENF	/(413)	BNTG	I	TPH1
						GETIT	I	TPH1
						SDINP	O	TPH1
						TEST	O	TPH1
						TOPM	I	TPH1
TLS1		Arc end times for nominal trajectory		/GENF	/(433)	BNTG	I	TST1
						GETIT	I	TST1
						PROPIN	I	TST1
						SDINP	O	TST1
						TEST	O	TST1
						TOPM	I	TST1
						TRAN3	I	TST1
						TRTOSZ	I	TST1

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DIP1		Phase initial times for nominal trajectory [sd]		/GENF	/(453)	GETIT	I	DIP1
						SDINP	M	DIP1
						TEST	O	DIP1
						TOPM	I	DIP1
DIS1		Arc initial times for nominal trajectory [sd]		/GENF	/(473)	GETIT	I	DIS1
						SDINP	M	DIS1
						TEST	O	DIS1
						TOPM	I	DIS1
						TRAN3	I	DIS1
TIME	t	Time (elapsed)		/GENF	/(493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CDW3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						VREF3	M	TIME
OMP		Phase cut-off value		/GENF	/(494)	ADJUST	O	OMP
						FNTG	M	OMP
						STP3	I	OMP
TIMPR		Trajectory print time		/GENF	/(495)	BNTG	O	TIMPR
						FNTG	M	TIMPR
						RKTA3A	I	TP
						RKTB3A	I	TP
LIFT	L	Aerodynamic lift	(LBS)	/GENF	/(496)	ACCEL	I	LIFT
						BL4	I	LIFT
						BL5	I	LIFT
						BL6	I	LIFT
						ENVPRM	I	LIFT
						FH3	I	LIFT
						OUT	I	LIFT
						PROPB	O	LIFT
						PROPIN	O	LIFT
						VT	O	LIFT
DRAG	D	Aerodynamic drag	(LBS)	/GENF	/(497)	ACCEL	I	DRAG
						BL5	I	DRAG
						BL7	I	DRAG
						BL8	I	DRAG
						ENVPRM	I	DRAG
						FH3	I	DRAG
						OUT	I	DRAG
						PROPB	O	DRAG
						PROPIN	O	DRAG
						SDER3	I	DRAG
						VT	M	DRAG
TBURN	t_b	Rocket burn initiation time on forward trajectory[sd]		/GENF	/(499)	EQUA3	I	TBURN
						MODELA	I	TBURN
						PROPB	O	TBURN
						PROPIN	M	TBURN
TBU		Saved rocket burn initiation times used during adjoint integration (sd)		/GENF	/(500)	PROPB	I	TBU
						PROPIN	O	TBU
AE	A_{exit}	Total nozzle exit area		/GENF	/(520)	ACCEL	I	AE
						FH2	I	AE
						IMPUL	I	AE
						PROPB	O	AE
						PROPIN	O	AE
						SDER3	I	AE

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
FP		Current value of cut-off function - non-linear only	/GENF	/(521)	DTF3	I	FP
					STP3	I	FP
					TOL3	I	FP
					YREF3	I	FP
FPOLD		Value of non-linear cut-off function at prior compute interval	/GENF	/(522)	DTF3	I	FPOLD
					STP3	O	FPOLD
					TOL3	O	FPOLD
					YREF3	I	FPOLD
FPD		Rate of change of non-linear cut-off function	/GENF	/(523)	CON3	I	FPD
					DTF3	I	FPD
					STP3	I	FPD
					YREF3	I	FPD
MACHR		Partial of mach number wrt altitude	/GENF	/(524)	EQUA3	O	MACHR
					PROPB	O	ZERO
					PROPIN	O	ZERO
					VT	I	MACHR
MACHV		Partial of mach number wrt velocity	/GENF	/(525)	EQUA3	O	MACHV
					VT	I	MACHV
QR		Partial of dynamic pressure wrt altitude	/GENF	/(526)	EQUA3	M	QR
					VT	I	QR
QV		Partial of dynamic pressure wrt velocity	/GENF	/(527)	EQUA3	M	QV
					TEST	M	QV
					VT	I	QV
FVAC		Total vacuum thrust [rocket] (LBS)	/GENF	/(528)	ACCEL	I	FVAC
					EQUA3	M	FVAC
					FH2	I	FVAC
					IMPUL	M	FVAC
					PROPB	M	FVAC
					PROPIN	M	FVAC
					SDER3	I	FVAC
LIFTV		Partial of lift wrt velocity	/GENF	/(529)	ACCEL	I	LIFTV
					BL4	I	LIFTV
					BL5	I	LIFTV
					BL6	I	LIFTV
					FH3	I	LIFTV
					VT	O	LIFTV
LIFTR		Partial of lift wrt altitude	/GENF	/(530)	ACCEL	I	LIFTR
					BL4	I	LIFTR
					BL5	I	LIFTR
					BL6	I	LIFTR
					FH3	I	LIFTR
					VT	O	LIFTR
LIFTA		Partial of lift wrt angle-of-attack	/GENF	/(531)	ACCEL	I	LIFTA
					BL4	I	LIFTA
					BL5	I	LIFTA
					BL6	I	LIFTA
					FH3	I	LIFTA
					VT	O	LIFTA
DRAGV		Partial of drag wrt velocity	/GENF	/(532)	ACCEL	I	DRAGV
					BL5	I	DRAGV
					BL7	I	DRAGV
					BL8	I	DRAGV
					FH3	I	DRAGV
					VT	M	DRAGV
DRAGR		Partial drag wrt altitude	/GENF	/(533)	ACCEL	I	DRAGR
					BL5	I	DRAGR
					BL7	I	DRAGR
					BL8	I	DRAGR
					FH3	I	DRAGR
					VT	M	DRAGR
DRAGA		Partial of drag wrt angle of attack	/GENF	/(534)	ACCEL	I	DRAGA
					BL5	I	DRAGA
					BL7	I	DRAGA
					BL8	I	DRAGA
					FH3	I	DRAGA
					VT	M	DRAGA

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
LIFTM		Partial of LIFT wrt mass		/GENF	/(535)	ACCEL	I	LIFTM
						BL4	I	LIFTM
						BL5	I	LIFTM
						BL6	I	LIFTM
						FH3	I	LIFTM
						VT	O	LIFTM
DBR		Partial of base drag wrt altitude		/GENF	/(536)	ACCEL	I	DBR
						BL4	I	DBR
						BL6	I	DBR
						BL7	I	DBR
						BL8	I	DBR
						EQUA3	I	DBR
						FH3	I	DBR
						VT	I	DBR
DB	D_b	Base drag	(LBS)	/GENF	/(537)	ACCEL	I	DB
						BL4	I	DB
						BL6	I	DB
						BL7	I	DB
						BL8	I	DB
						EQUA3	I	DB
						FH3	I	DB
						OUT	I	DB
						SDER3	I	DB
						VT	I	DB
ISP	I_{sp}	Net vacuum specific impulse		/GENF	/(538)	ACCEL	I	ISP
						IMPUL	O	ISP
ISPF		Partial of ISP wrt vacuum thrust		/GENF	/(539)	ACCEL	I	ISPF
						IMPUL	O	ISPF
ULFT	L_U	Untrimmed aero.lift		/GENF	/(540)	BL3	I	ULFT
						MODELA	I	ULFT
						VT	M	ULFT
ULFTV		Partial of ULFT wrt velocity		/GENF	/(541)	BL3	I	ULFTV
						VT	M	ULFTV
ULFTR		Partial of ULFT wrt altitude		/GENF	/(542)	BL3	I	ULFTR
						VT	M	ULFTR
ULFTA		Partial of ULFT wrt angle of attack		/GENF	/(543)	BL3	I	ULFTA
						VT	M	ULFTA
XMCG	M_{CG}	Aerodynamic moment about center of gravity	(FT-LBS)	/GENF	/(544)	EL2	I	XMCG
						OUT	I	XMCG
						VT	M	XMCG
XMCGV		Partial of XMCG wrt velocity		/GENF	/(545)	EL2	I	XMCGV
						VT	M	XMCGV
XMCGR		Partial of XMCG wrt altitude		/GENF	/(546)	EL2	I	XMCGR
						VT	M	XMCGR
XMCGA		Partial of XMCG wrt angle of attack		/GENF	/(547)	EL2	I	XMCGA
						VT	M	XMCGA
XMCGM		Partial of XMCG wrt mass		/GENF	/(548)	EL2	I	XMCGM
						VT	M	XMCGM
CODAE	$\cos(\alpha-\delta_E)$	See symbol		/GENF	/(549)	ACCEL	I	CODAE
						BL4	I	CODAE
						BL6	I	CODAE
						BL7	I	CODAE
						BL8	I	CODAE
						FH3	I	CODAE
						SDER3	I	CODAE
						VT	O	CODAE
CULFT		Constant value of ULFT	(LBS)	/GENF	/(550)	BL3	I	CULFT
						MODELA	O	CULFT
CT		Constant value of vacuum thrust	(LBS)	/GENF	/(551)	FH1	I	CT
CALPHA		Constant value of angle-of-attack	(RAD)	/GENF	/(552)	BL2	I	CALPHA
						MODELA	M	CALPHA
						MODELB	M	CALPHA

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CDE		Constant value of engine deflection (RAD)	/GENF	/(553)	EL1	I	CDE
					PROPB	O	CDE
					PROPIN	O	CDE
DELTAE	δ_E	Engine gimbal deflection angle (RAD)	/GENF	/(554)	BLGCON	M	DELTAE
					EL1	I	DELTAE
					OUT	I	DELTAE
					REU3	O	DELTAE
					VT	I	DELTAE
SID	$\sin(\delta_E)$	See symbol	/GENF	/(555)	EL2	I	SID
					OUT	I	SID
					VT	M	SID
COD	$\cos(\delta_E)$	See symbol	/GENF	/(556)	EL2	I	COD
					OUT	I	COD
					VT	M	COD
SIDAE	$\sin(\alpha-\delta_E)$	See symbol	/GENF	/(557)	ACCEL	I	SIDAE
					BL4	I	SIDAE
					BL6	I	SIDAE
					BL7	I	SIDAE
					BL8	I	SIDAE
					FH3	I	SIDAE
					VT	O	SIDAE
XCG	X_{CG}	Center of gravity body x station (FT)	/GENF	/(558)	EL2	I	XCG
					EQUA3	I	XCG
					VT	I	XCG
ZCG	Z_{CG}	Center of gravity body z station (FT)	/GENF	/(559)	EL2	I	ZCG
					EQUA3	I	ZCG
					VT	I	ZCG
XJ	J	Control blend factor	/GENF	/(560)	EL2	I	XJ
					EQUA3	I	XJ
					OUT	I	XJ
					VT	I	XJ
XJV		Partial of blend factor wrt velocity	/GENF	/(561)	EL2	I	XJV
					EQUA3	O	XJV
					VT	I	XJV
XJR		Partial of blend factor wrt altitude	/GENF	/(562)	EL2	I	XJR
					EQUA3	O	XJR
					VT	I	XJR
GH		Partial of gravity wrt altitude	/GENF	/(563)	BL7	I	GH
					BL8	I	GH
					EQUA3	O	GH
					PDY3A	I	DGDH
GAMMAD		Pitch rate (RAD)	/GENF	/(564)	BL4	I	GAMMAD
					PROPB	O	GAMMAD
					PROPIN	O	GAMMAD
XKG	k_γ	Algebraic equation used in vertical rise and pitchover	/GENF	/(565)	BL4	I	XKG
					MODELA	M	XKGAM
					MODELB	M	XKGAM
XKP	k_ψ	Algebraic equation used in vertical rise and pitchover	/GENF	/(566)	BL4	I	XKP
					MODELA	M	XKPS
					MODELA	I	XKPSI
					MODELB	M	XKPS
					MODELB	I	XKPSI
FRATED		Net rated maximum rocket vacuum thrust (LBS)	/GENF	/(567)	IMPUL	I	FRATED
					PROPB	O	FRATED
					PROPIN	O	FRATED
IRATED		Maximum rated ISP (SEC)	/GENF	/(568)	IMPUL	I	IRATED
P1		First element in in-plane control vector. Corresponds to thrust	/GENF	/(569)	BLGCON	M	P
					BLGCON	M	P1
					OUT	I	P1
P2		Second element in in-plane control vector. Corresponds to deflection	/GENF	/(570)	BLGCON	M	P2
P3		Third element in in-plane control vector, corresponds to α	/GENF	/(571)	BLGCON	M	P3

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
XK1		First control vector governing equation value corresponds to error in thrust	/GENF	/(572)	BLGCON	I	XK1
					FH1	O	XK1
					FH2	O	XK1
					FH3	O	XK1
					FH4	O	XK1
					MODELA	I	XK1
XK2		Second control vector governing equation value. If non-trivial corresponds to error in moment balance	/GENF	/(573)	EL1	O	XK2
					EL2	O	XK2
XK3		Third control vector governing equation value. Corresponds to error in algebraic equation involving α.	/GENF	/(574)	BL2	O	XK3
					BL3	O	XK3
					BL4	O	XK3
					BL5	O	XK3
					BL6	O	XK3
					BL7	O	XK3
					BL8	O	XK3
					MODELA	I	XK3
					OUT	I	XK3
XK1T		Partial of governing equation wrt state or control vector component	/GENF	/(575)	BLGCON	I	XK1T
					FH1	O	XK1T
					FH2	O	XK1T
					FH3	M	XK1T
					FH4	O	XK1T
					MODELA	I	XK1T
XK2T		Partial of governing equation wrt state or control vector component	/GENF	/(576)	BLGCON	I	XK2T
					EL2	O	XK2T
XK3T		Partial of governing equation wrt state or control vector component	/GENF	/(577)	BLGCON	I	XK3T
					BL4	O	XK3T
					BL6	O	XK3T
					BL7	O	XK3T
					BL8	O	XK3T
XK1D		Partial of governing equation wrt state or control vector component	/GENF	/(578)	BLGCON	I	XK1D
					FH3	M	XK1D
XK2D		Partial of governing equation wrt state or control vector component	/GENF	/(579)	ACCEL	I	XK2D
					BLGCON	I	XK2D
					EL1	O	XK2D
					EL2	O	XK2D
XK3D		Partial of governing equation wrt state or control vector component	/GENF	/(580)	BLGCON	I	XK3D
					BL4	O	XK3D
					BL6	O	XK3D
					BL7	O	XK3D
					BL8	O	XK3D
XK1A		Partial of governing equation wrt state or control vector component	/GENF	/(581)	BLGCON	I	XK1A
					FH3	M	XK1A
XK2A		Partial of governing equation wrt state or control vector component	/GENF	/(582)	ACCEL	I	XK2A
					BLGCON	I	XK2A
					EL2	O	XK2A
XK3A		Partial of governing equation wrt state or control vector component	/GENF	/(583)	BLGCON	I	XK3A
					BL2	O	XK3A
					BL3	O	XK3A
					BL4	O	XK3A
					BL5	M	XK3A
					BL6	O	XK3A
					BL7	O	XK3A
					BL8	O	XK3A
XK1V		Partial of governing equation wrt state or control vector component	/GENF	/(584)	BLGCON	I	XK1V
					FH3	M	XK1V
					FH4	O	XK1V
XK2V		Partial of governing equation wrt state or control vector component	/GENF	/(585)	EL2	O	XK2V
XK3V		Partial of governing equation wrt state or control vector component	/GENF	/(586)	BL3	O	XK3V
					BL4	O	XK3V
					BL5	M	XK3V
					BL6	O	XK3V
					BL7	O	XK3V
					BL8	O	XK3V

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
XK1G		Partial of governing equation wrt state or control vector component	/GENF	/(587)			
XU2G		Partial of governing equation wrt state or control vector component	/GENF	/(588)			
XK3G		Partial of governing equation wrt state or control vector component	/GENF	/(589)	BL4	O	XK3G
					BL7	O	XK3G
					BL8	O	XK3G
XK1P		Partial of governing equation wrt state or control vector component	/GENF	/(590)			
XK2P		Partial of governing equation wrt state or control vector component	/GENF	/(591)			
XK3P		Partial of governing equation wrt state or control vector component	/GENF	/(592)	BL4	O	XK3P
					BL7	O	XK3P
					BL8	O	XK3P
XK1R		Partial of governing equation wrt state or control vector component	/GENF	/(593)	FH2	O	XK1R
					FH3	M	XK1R
					FH4	O	XK1R
XK2R		Partial of governing equation wrt state or control vector component	/GENF	/(594)	EL2	O	XK2R
XK3R		Partial of governing equation wrt state or control vector component	/GENF	/(595)	BL3	O	XK3R
					BL4	O	XK3R
					BL5	M	XK3R
					BL6	O	XK3R
					BL7	O	XK3R
					BL8	O	XK3R
XK1O		Partial of governing equation wrt state or control vector component	/GENF	/(596)			
XK2O		Partial of governing equation wrt state or control vector component	/GENF	/(597)			
XK3O		Partial of governing equation wrt state or control vector component	/GENF	/(598)	BL4	O	XK3O
					BL7	O	XK3O
					BL8	O	XK3O
XK1U		Partial of governing equation wrt state or control vector component	/GENF	/(599)			
XK2U		Partial of governing equation wrt state or control vector component	/GENF	/(600)			
XK3U		Partial of governing equation wrt state or control vector component	/GENF	/(601)			
XK1M		Partial of governing equation wrt state or control vector component	/GENF	/(602)	FH3	M	XK1M
XK2M		Partial of governing equation wrt state or control vector component	/GENF	/(603)	EL2	O	XK2M
XK3M		Partial of governing equation wrt state or control vector component	/GENF	/(604)	BL4	O	XK3M
					BL5	M	XK3M
					BL6	O	XK3M
					BL7	O	XK3M
					BL8	O	XK3M
PV		Partial of ϕ wrt state	/GENF	/(605)	BL4	M	PV
PG		Partial of ϕ wrt state	/GENF	/(606)	BL4	M	PG
PP		Partial of ϕ wrt state	/GENF	/(607)	BL4	M	PP
PR		Partial of ϕ wrt state	/GENF	/(608)	BL4	M	PR
PO		Partial of ϕ wrt state	/GENF	/(609)	BL4	M	PO
DPDY	$\partial w/\partial y$	Matrix of partials of in-plane control vector wrt state	/GENF	/(610)	ACCEL	I	DPDY
					BLGCON	I	DPDY
					OUT	I	DPDY

BLØCK
PARAM

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IPOINT		Code for each adjustable parameter in steepest descent	/PARAM /(	1)	ADJUST	I	IPOINT
					PRMSET	I	IPOINT
					SDINP	O	IPOINT
					STAU	I	IPOINT
					TOPM	D	IPOINT
NPARA		Number of adjustable parameters in trajectory problem	/PARAM /(	13)	ADJUST	I	NPARA
					BNTG	I	NPARA
					FNTG	I	NPARA
					MTX3A	I	NPARA
					PAYO2	I	NPARA
					PRMSET	I	NPARA
					SDINP	M	NPARA
					STAU	I	NPARA
					TEST	I	NPARA
					TOPM	D	NPARA
NPA		Running count of number of adjustable parameters to be perturbed on remainder of trajectory	/PARAM /(	14)	ADJUST	M	NPAR
					FNTG	I	NPA
					MTX3A	I	NPA
					TOPM	D	NPA
SPARA	$S^{\Psi}{}_{i}$	Matrix of adjustable parameter sensitivities (including all parameters)	/PARAM /(	15)	ADJUST	I	SPARA
					PAYO2	I	SPARA
					STAU	M	SPARA
					TOPM	D	SPARA
WTPD		Input or preset adjustable parameter weighting factors according to type of parameter	/PARAM /(	123)	SDINP	M	WTPD
					TOPM	D	WTPD
WTP	[Y]	Adjustable parameter diagonal weighting matrix order according to IPOINT.	/PARAM /(	132)	MTX3A	I	WTP
					PAYO2	I	WTP
					SDINP	O	WTP
					TOPM	D	WTP
SPARB	$S^{\Psi}{}_{r}$	Matrix of adjustable parameter sensitivities. (Contains only elements corresponding to parameters yet to be adjusted)	/PARAM /(	144)	ADJUST	O	SPARB
					MTX3A	I	SPARB
					TOPM	D	SPARB
PARA	p	Adjustable parameter nominal values.	/PARAM /(	252)	ADJUST	I	PARA
					PRMSET	M	PARA
					TOPM	D	PARA
DPAR	δp	Adjustable parameter corrections	/PARAM /(	264)	ADJUST	I	DPAR
					MTX3A	M	DPAR
					TOPM	D	DPAR
S2INV	[SS]	Parameter sensitivity contribution to A matrix	/PARAM /(	276)	ADJUST	O	S2INV
					MTX3A	M	S2INV
					PAYO2	M	S2INV
					TOPM	D	S2INV
DELP		Input or preset nominal parameter adjustment size	/PARAM /(	357)	SDINP	O	DELP
					TOPM	D	DELP

BLØCK
RETREV

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
FTIME		Time at which trajectory data set is stored. (SEC)	/RETREV/(	1)	AGETB3 AST3 GETIT TRAN3	O O I I	FTIME FTIME FTIME FTIME
BTIME		Time at which adjoint data set is stored. (SEC)	/RETREV/(	2)			
MAXA		Number of words in last stored partial buffer of trajectory data MAXA(1) corresponds to random file 39 MAXA(2) corresponds to random file 40.	/RETREV/(	3)	AGETB3 AST3	I M	MAXA MAXA
MAXB		Number of words in last stored partial buffer of adjoint data. Corresponds to random file 41.	/RETREV/(	5)	BGET3 BSTO3	I M	MAXB MAXB
NBUFA		Number of buffers of trajectory data stored on random files 39 and 40 respectively.	/RETREV/(	6)	AGETB3 AST3	I M	NBUFA NBUFA
IBUF1		Counts number of buffers as forward trajectory is stored on random file.	/RETREV/(	8)	AST3	M	IBUF1
IBUF2		Counts number of buffers as forward trajectory data is retrieved from random file.	/RETREV/(	9)	AGETB3 AST3	M M	IBUF2 IBUF2
NBFA		Maximum number of buffers permitted to store forward trajectory data = 20.	/RETREV/(	10)	AST3 TOPM	I D	NBFA NBFA
NBFB		Maximum number of buffers permitted to store adjoint solution data = 60.	/RETREV/(	11)	BSTO3 TOPM	I D	NBFB NBFB
MIXA		Maximum number of words in trajectory data buffer = 990.	/RETREV/(	12)	AGETB3 AST3 SDINP TOPM	I I I D	MIXA MIXA MIXA MIXA
MIXB		Maximum number of words in adjoint data buffer = 3000	/RETREV/(	13)	BGET3 BSTO3 SDINP TOPM	I I I D	MIXB MIXB MIXB MIXB
MXA		Index of last stored word in full buffer of forward trajectory data.	/RETREV/(	14)	AGETB3 AST3 SDINP	I I D	MXA MXA MXA
MXB		Index of last stored word in full buffer of adjoint data.	/RETREV/(	15)	BGET3 BSTO3 SDINP	I I D	MXB MXB MXB
NPTA		Number of words stored at each trajectory time point.	/RETREV/(	16)	SDINP	M	NPTA
NPTB		Number of words stored at each adjoint solution time point.	/RETREV/(	17)	BGET3 SDINP	I M	NPTB NPTB
IBLKB		Index of adjoint data buffer where either last word was stored or retrieved.	/RETREV/(	18)	BGET3	M	IBLKB
NBUFB		Number of buffers of adjoint data stored on last adjoint solution.	/RETREV/(	19)	BGET3 BSTO3	I D	NBUFB NBUFB
IBUFB		Counts number of buffers of adjoint data that have either been stored or retrieved as solution progress.	/RETREV/(	20)	BGET3 BSTO3	M M	IBUFB IBUFB

BLØCK
SPECØ

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
UMU2	$\partial^2\nu/\partial R^2$	~~Description not input~~ SEE SYMBOL	/SPECO /(	1)	ANLATM PAT63	O O	UMU2 UMU2
RORR	$\partial^2\rho_a/\partial R^2$	~~Description not input~~ SEE SYMBOL	/SPECO /(	2)	ANLATM BL7 BL8 PAT63	O I I O	RORR RORR RORR RORR

30 OCT 72 G.01-46

BLØCK
STATE3

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
V	v	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V
VAR	y	State vector in steepest descent module		/STATE3/(	1)			
GAM	γ	Relative flight path angle	(RAD)	/STATE3/(	2)	EQUA3	I	GAM
						GUI3A	I	GAM
						OUT	I	GAM
ALT	h	Altitude		/STATE3/(	3)	EQUA3	I	ALT
						OUT	I	ALT
						PDBC	I	ALT
M	m	Mass		/STATE3/(	4)	ACCEL	I	M
						BL4	I	M
						BL8	I	M
						EQUA3	I	M
						OUT	I	M
						SDER3	I	M
PSI	ψ	Azimuth		/STATE3/(	5)	EQUA3	I	PSI
						GUI3A	I	PSI
						OUT	I	PSI
RHO	ρ	Latitude		/STATE3/(	6)	EQUA3	I	RHO
						OUT	I	RHO
MU	μ	Longitude		/STATE3/(	7)	OUT	I	MU
						PDBC	I	MU
HT	Q	Heating		/STATE3/(	8)	OUT	I	HT
DVAR	$\dot{y}$	State vector derivatives in steepest descent module		/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
VD	$\dot{V}$	V derivative		/STATE3/(	15)			
GD	$\dot{\gamma}$	GAM derivative		/STATE3/(	16)	DER3A	O	GD
						PDBC	I	GD
HD	$\dot{h}$	ALT derivative		/STATE3/(	17)	DER3A	O	HD
						PDBC	I	HD

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
MD	$\dot{m}$	MASS derivative	/STATE3/(	18)	DER3A OUT	O I	MD MD
PD	$\dot{\psi}$	Azimuth derivative	/STATE3/(	19)	DER3A	O	PSID
OD	$\dot{\rho}$	Latitude derivative	/STATE3/(	20)	DER3A	O	OD
UD	$\dot{\mu}$	Longitude derivative	/STATE3/(	21)	DER3A PDBC	O I	UD UD
HTD	$\dot{Q}$	Heating derivative	/STATE3/(	22)	DER3A OUT PDBC PDY3A	O I I M	HTD HTD HTD HTD
VARL		Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A ADICB3 ADIC3A PROPIN RKTA3A RKTB3A STVRL3 TRAN3	I O O I M M O M	VARL VARL VARL ZZ F Y VARL VARL
QMAX1		Description not input	/STATE3/(	39)	PROPIN	I	QMAX1
HDMAX1		Description not input	/STATE3/(	42)	PROPIN	I	HDMAX1
REMAX1		Description not input	/STATE3/(	69)	PROPIN	I	REMAX1
DELY		Description not input	/STATE3/(	71)	RKTA3A	M	DELY
DVARL		Array of derivatives for adjoint integration	/STATE3/(	128)	ADEQ3A ADICB3 ADIC3A RKTB3A	O O O I	DVARL DVARL DVARL DY
RR		Description not input	/STATE3/(	157)	MTX3A	M	RR
DELY		Description not input	/STATE3/(	167)	MTX3A	M	DELY
JJ		Description not input	/STATE3/(	177)	MTX3A	M	JJ
YO	y_{old}	State vector of nominal trajectory	/STATE3/(	227)	AST3 MTX3A	O I	YO YO
SVY	y_{-1}	State and time array at previous compute interval	/STATE3/(	236)	DTF3 REU3 YREF3	I M I	SVY SVY SVY
XL	$\lambda^{\Psi_i \Omega_j}$	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A ADICB3 ADIC3A ADID3A AST3 BGET3 BST03 MTX3A OUT STAU STVRL3 TRAN3	M M M M M O I I I M I M	XL XL XL XL XL XL XL XL XL XL XL XL
YDP		Array of state derivatives at phase end points	/STATE3/(	327)	ADID3A REU3	I O	YDP YDP
YDS	$\dot{y}\|_{\tau-}$	Array of state derivatives at arc end points	/STATE3/(	507)	ADICB3 ADID3A REU3 STAU	I I O I	YDS YDS YDS YDS
COSGAM	$\cos(\gamma)$	See symbol	/STATE3/(	687)	ACCEL BL4 BL8 DER3A EQUA3 MODELA MODELB OUT PDBC PDY3A	I I I I O I I I I I	COSGAM COSGAM COSGAM COSGAM COSGAM COSGAM COSGAM COSGAM COSGAM COSGAM

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SINGAM	$\sin(\gamma)$	See symbol	/STATE3/(	688)	BL4	I	SINGAM
					BL7	I	SINGAM
					BL8	I	SINGAM
					DER3A	I	SINGAM
					EQUA3	O	SINGAM
					MODELA	I	SINGAM
					MODELB	I	SINGAM
					PDBC	I	SINGAM
					PDY3A	I	SINGAM
					SDER3	I	SINGAM
SAVBP		Saved state vector at branching point for initializing second brnch	/STATE3/(	689)	REU3	M	SAVBP
SINPSI	$\sin(\psi)$	See symbol	/STATE3/(	704)	BL4	I	SINPSI
					BL7	I	SINPSI
					BL8	I	SINPSI
					DER3A	I	SINPSI
					EQUA3	O	SINPSI
					MODELA	I	SINPSI
					MODELB	I	SINPSI
					PDBC	I	SINPSI
					PDY3A	I	SINPSI
COSPSI	$\cos(\psi)$	See symbol	/STATE3/(	705)	BL4	I	COSPSI
					BL7	I	COSPSI
					BL8	I	COSPSI
					DER3A	I	COSPSI
					EQUA3	O	COSPSI
					MODELA	I	COSPSI
					MODELB	I	COSPSI
					PDBC	I	COSPSI
					PDY3A	I	COSPSI
SINRHO	$\sin(\rho)$	See symbol	/STATE3/(	706)	BL4	I	SINRHO
					BL7	I	SINRHO
					BL8	I	SINRHO
					DER3A	I	SINRHO
					EQUA3	O	SINRHO
					MODELA	I	SINRHO
					MODELB	I	SINRHO
					OUT	I	SINRHO
					PDBC	I	SINRHO
					PDY3A	I	SINRHO
COSRHO	$\cos(\rho)$	See symbol	/STATE3/(	707)	BL4	I	COSRHO
					BL7	I	COSRHO
					BL8	I	COSRHO
					DER3A	I	COSRHO
					EQUA3	M	COSRHO
					MODELA	I	COSRHO
					MODELB	I	COSRHO
					OUT	I	COSRHO
					PDBC	I	COSRHO
					PDY3A	I	COSRHO
OCORHO	$\omega\times$COSRHO	See symbol	/STATE3/(	708)	DER3A	I	OCORHO
					EQUA3	M	OCORHO
					MODELA	I	OCORHO
					MODELB	I	OCORHO
					PDBC	I	OCORHO
					PDY3A	I	OCORHO
OCORO2	$\omega\times$OCORHO	See symbol	/STATE3/(	709)	DER3A	I	OCORO2
					EQUA3	O	OCORO2
					MODELA	I	OCORO2
					MODELB	I	OCORO2
					PDY3A	I	OCORO2
SVBV		Saved state vector on trial trajectory	/STATE3/(	710)	ADICB3	I	SVBV
					REU3	O	SVBV
OMEGA	ω	Earth rotation rate (RAD/SEC)	/STATE3/(	719)	BL4	I	OMEGA
					BL7	I	OMEGA
					TOPM	O	OMEGA

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
OMEGA2	ω^2	See symbol	/STATE3/(	720)	BL4 BL7 BL8 TOPM	I I I O	OMEGA2 OMEGA2 OMEGA2 OMEGA2
VDV		Partial derivative of equation of motion WRT state	/STATE3/(	721)	ADEQ3A PDY3A	I O	VDV VDV
GDV		Partial derivative of equation of motion WRT state	/STATE3/(	722)	ADEQ3A PDY3A	I O	GDV GDV
RDV		Partial derivative of equation of motion WRT state	/STATE3/(	723)	ADEQ3A PDY3A	I O	RDV RDV
MDV		Partial derivative of equation of motion WRT state	/STATE3/(	724)	ADEQ3A PDY3A	I O	MDV MDV
PDV		Partial derivative of equation of motion WRT state	/STATE3/(	725)	ADEQ3A PDY3A	I O	PDV PDV
ODV		Partial derivative of equation of motion WRT state	/STATE3/(	726)	ADEQ3A PDY3A	I M	ODV ODV
UDV		Partial derivative of equation of motion WRT state	/STATE3/(	727)	ADEQ3A PDY3A	I M	UDV UDV
VDG		Partial derivative of equation of motion WRT state	/STATE3/(	728)	ADEQ3A PDY3A	I O	VDG VDG
GDG		Partial derivative of equation of motion WRT state	/STATE3/(	729)	ADEQ3A PDY3A	I O	GDG GDG
RDG		Partial derivative of equation of motion WRT state	/STATE3/(	730)	ADEQ3A PDY3A	I O	RDG RDG
PDG		Partial derivative of equation of motion WRT state	/STATE3/(	731)	ADEQ3A PDY3A	I O	PDG PDG
ODG		Partial derivative of equation of motion WRT state	/STATE3/(	732)	ADEQ3A PDY3A	I O	ODG ODG
UDG		Partial derivative of equation of motion WRT state	/STATE3/(	733)	ADEQ3A PDY3A	I O	UDG UDG
VDR		Partial derivative of equation of motion WRT state	/STATE3/(	734)	ADEQ3A PDY3A	I O	VDR VDR
GDR		Partial derivative of equation of motion WRT state	/STATE3/(	735)	ADEQ3A PDY3A	I O	GDR GDR
MDR		Partial derivative of equation of motion WRT state	/STATE3/(	736)	ADEQ3A PDY3A	I O	MDR MDR
PDR		Partial derivative of equation of motion WRT state	/STATE3/(	737)	ADEQ3A PDY3A	I O	PDR PDR
ODR		Partial derivative of equation of motion WRT state	/STATE3/(	738)	ADEQ3A PDY3A	I O	ODR ODR
UDR		Partial derivative of equation of motion WRT state	/STATE3/(	739)	ADEQ3A PDY3A	I M	UDR UDR
VDM		Partial derivative of equation of motion WRT state	/STATE3/(	740)	ADEQ3A PDY3A	I O	VDM VDM
GDM		Partial derivative of equation of motion WRT state	/STATE3/(	741)	ADEQ3A PDY3A	I O	GDM GDM
MDM		Partial derivative of equation of motion WRT state	/STATE3/(	742)	ADEQ3A PDY3A	I O	MDM MDM
PDM		Partial derivative of equation of motion WRT state	/STATE3/(	743)	ADEQ3A PDY3A	I O	PDM PDM
VDP		Partial derivative of equation of motion WRT state	/STATE3/(	744)	ADEQ3A PDY3A	I O	VDP VDP
GDP		Partial derivative of equation of motion WRT state	/STATE3/(	745)	ADEQ3A PDY3A	I O	GDP GDP
PDP		Partial derivative of equation of motion WRT state	/STATE3/(	746)	ADEQ3A PDY3A	I O	PDP PDP
ODP		Partial derivative of equation of motion WRT state	/STATE3/(	747)	ADEQ3A PDY3A	I O	ODP ODP

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
UDP		Partial derivative of equation of motion WRT state	/STATE3/(	748)	ADEQ3A PDY3A	I O	UDP UDP
VDO		Partial derivative of equation of motion WRT state	/STATE3/(	749)	ADEQ3A PDY3A	I O	VDO VDO
GDO		Partial derivative of equation of motion WRT state	/STATE3/(	750)	ADEQ3A PDY3A	I O	GDO GDO
PDO		Partial derivative of equation of motion WRT state	/STATE3/(	751)	ADEQ3A PDY3A	I O	PDO PDO
UDO		Partial derivative of equation of motion WRT state	/STATE3/(	752)	ADEQ3A PDY3A	I O	UDO UDO
HTDV		Partial derivative of equation of motion WRT state	/STATE3/(	753)	ADEQ3A PDY3A	I O	HTDV HTDV
HTDR		Partial derivative of equation of motion WRT state	/STATE3/(	754)	ADEQ3A PDY3A	I O	HTDR HTDR
SIN2RO	$\sin(2\rho)$	See symbol	/STATE3/(	755)	BL4 BL7 BL8 MODELA MODELB	I M M O O	SIN2RO SIN2RO SIN2RO SIN2RO SIN2RO
COS2RO	$\cos(2\rho)$	See symbol	/STATE3/(	756)	BL4 BL7 BL8 MODELA MODELB	I M M O O	COS2RO COS2RO COS2RO COS2RO COS2RO
COS2GM	$\cos(2\gamma)$	See symbol	/STATE3/(	757)	MODELA MODELB	O O	COS2GM COS2GM

BLØCK
STS

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DPAY	$d\Phi$	Initial payoff improvement	/STS	/(1)	PAY02	I	DPAY
					SDINP	I	DPAY
					SDINP	O	IST
					SDINP	I	ST
					TEST	I	DPAY
					TOPM	O	IDPAY
PMIN		Minimum payoff improvement	/STS	/(2)	PAY02	I	PMIN
					SDINP	I	PMIN
					TEST	I	PMIN
WORK		Working array, contains TOPEN1, TOPEN2, and PHIWT	/STS	/(3)	ADEQ3A	I	WORK
					FNTG	I	WORK
					MODELB	O	WORK
					MTX3A	I	WORK
					SDINP	M	WORK
					TEST	M	WORK
DQ		~~Description not input~~ SAVED PAYOFF IMPROVEMENT	/STS	/(7)	PAY02	M	DQ
CMULT		~~Description not input~~ NON-OPT. CONTROL MULTIPLIER	/STS	/(13)	ADEQ3A	I	CMULT
NITER		Maximum number of steepest descent iterations	/STS	/(31)	SDINP	I	NITER
					TEST	I	NITER
MNGA		Control history curve number	/STS	/(32)	GUI3A	M	MNGA
					SDINP	M	MNGA
					SDINP	I	TAL
MNGP		Control history curve number	/STS	/(72)	GUI3A	M	MNGP
					SDINP	M	MNGP
					SDINP	I	WTPI
DLPI		Description not input	/STS	/(91)	SDINP	I	DLPI
AR		Array for storing starting control history tables	/STS	/(112)	SDINP	I	AR
					SDINP	I	III
					TBLK	I	AR
IAD		Starting address of each control history table	/STS	/(312)	SDINP	M	IAD
					TBLK	I	IAD
INP		Index of last argument of each control history table	/STS	/(332)	SDINP	M	INP
					TBLK	I	INP
ISV		Saved index of last control history table look-up	/STS	/(352)	SDINP	O	ISV
					TBLK	M	ISV

BLØCK XCØDES

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ITQ		Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
					ADIC3A	I	ITQ
					ADID3A	I	ITQ
					CON3	I	ITQ
					SDINP	M	ITQ
					STAU	I	ITQ
					TOPM	O	IITQ
ICOR		Phase sequence array	/XCODES/(	10)	ADJUST	I	ICOR
					FNTG	I	ICOR
					PRMSET	I	ICOR
					SDINP	M	ICOR
ITI		Optimized arc time flag	/XCODES/(	30)	ADJUST	M	ITI
					FNTG	I	ITI
					SDINP	O	ITI
INTB		Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
					BNTG	I	INTB
					ENVPRM	I	INTB
					FNTG	I	INTB
					SDINP	M	INTB
					TEST	I	INTB
					TRAN3	I	INTB
					TRTOSZ	I	INTB
JGID		Control option flag array	/XCODES/(	32)	BNTG	I	JGID
					FNTG	I	JGID
					SDINP	M	JGID
JPH		Phase cut-off option flag	/XCODES/(	72)	BNTG	I	JPH
					FNTG	M	JPH
					SDINP	M	JPH
JST		Arc cut-off option flag	/XCODES/(	112)	ADICB3	I	JST
					BNTG	I	JST
					FNTG	I	JST
					SDINP	M	JST
NCNST		Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
					BSTO3	I	NCNST
					CON3	I	NCNST
					PAYO2	I	NCNST
					SDINP	M	NCNST
					SUMS	I	NCNST
					TEST	I	NCNST
					TOPM	I	NCNST
					TRAN3	I	NCNST
NSB		Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
					BNTG	I	NSB
					ENVPRM	I	NSB
					FNTG	I	NSB
					REU3	I	NSB
					SDINP	M	NSB
					TEST	I	NSB
					TRAN3	I	NSB
					TRTOSZ	I	NSB
NSAB		Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
					BNTG	I	NSAB
					ENVPRM	I	NSAB
					FNTG	I	NSAB
					SDINP	M	NSAB
					TEST	I	NSAB
					TRAN3	I	NSAB
					TRTOSZ	I	NSAB
NICNB		Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
					ADIC3A	I	NICNB
					BNTG	I	NICNB
					REU3	I	NICNB
					SDINP	M	NICNB
					TEST	I	NICNB
					TRAN3	I	NICNB
I2OP		First optimization pass flag sets $d\Phi$ = DPAY. Also used to indicate payoff degradation due to restoration of constraints	/XCODES/(	136)	PAYO2	M	I2OP
					TEST	M	I2OP
					TOPM	O	I2OP

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ICOP		Flag used to indicate state derivative refinement at arc cut-off point.	/XCODES/(	137)	FNTG	M	ICOP
					GETIT	I	ICOP
IFAW		Logical file to write current trial trajectory data	/XCODES/(	138)	AST3	I	IFAW
					TOPM	M	IFAW
IFAR		File where nominal trajectory data is read from	/XCODES/(	139)	AGETB3	I	IFAR
					AST3	I	IFAR
					TOPM	M	IFAR
IFB		File where adjoint solution is stored	/XCODES/(	140)	BGET3	I	IFB
					BSTO3	I	IFB
					TOPM	O	IFB
IND		Flag indicates whether on first nominal trajectory (IND=1)	/XCODES/(	141)	AST3	I	IND
					BGET3	I	IND
					FNTG	M	IND
					GUI3A	I	IND
					MTX3A	I	IND
					PROPIN	I	IND
IOPEN		Closed to open-loop control switch when equal to 2	/XCODES/(	142)	FNTG	M	IOPEN
					PROPIN	O	IOPEN
IPH		Phase number	/XCODES/(	143)	ADID3A	I	IPH
					ADJUST	I	IPH
					AST3	I	IPH
					BNTG	M	IPH
					FNTG	M	IPH
					GETIT	I	IPH
					GUI3A	I	IPH
					SDINP	M	IPH
ISPH		Sign of phase cut-off	/XCODES/(	144)	FNTG	O	ISPH
					STP3	I	ISPH
ISST		Sign of arc cut-off	/XCODES/(	145)	FNTG	O	ISST
					STP3	I	ISST
IARC		Arc number	/XCODES/(	146)	ADICB3	I	IARC
					ADID3A	I	IARC
					ADJUST	I	IARC
					AST3	I	IARC
					BNTG	M	IARC
					ENVPRM	I	IARC
					FNTG	M	IARC
					GETIT	I	IARC
					MODELA	I	IARC
					PROPB	I	IARC
					PROPIN	I	IARC
					REU3	I	IARC
					SDINP	M	IARC
					STAU	I	IARC
					STP3	I	IARC
					TRTOSZ	I	IARC
ISTART		Initialization and divergance flag	/XCODES/(	147)	AST3	O	ISTART
					BLGCON	O	ISTART
					BLYNE	O	ISTART
					FNTG	I	ISTART
					MODELA	O	ISTART
					PROPIN	O	ISTART
					REU3	I	ISTART
					TEST	M	ISTART
					TOPM	M	ISTART
ITCT		Iteration counter	/XCODES/(	148)	BNTG	I	ITCT
					OUT	I	ITCT
					TEST	M	ITCT
					TOPM	M	ITCT

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ITER		Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
					FNTG	I	ITER
					GETIT	I	ITER
					MODELA	I	ITER
					OUT	I	ITER
					PAYO2	M	ITER
					PROPIN	I	ITER
					TEST	M	ITER
					TOPM	M	ITER
IVAR		Cut-off variable option indicator	/XCODES/(	150)	FNTG	M	IVAR
					STP3	O	IVAR
					TOL3	I	IVAR
JK		Integration routine flag tells which derivative evaluation in Runge-Kutta cycle	/XCODES/(	151)	ADIC3A	M	JK
					BNTG	I	JK
					MODELA	I	JK
					PAYO2	M	JK
					RKTA3A	M	J
					RKTB3A	M	J
JPS		Absolute value of phase cut-off option code	/XCODES/(	152)	ADID3A	I	JPS
					BNTG	M	JPS
					FNTG	M	JPS
					STP3	I	JPS
					TOL3	I	JPS
JS		Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
					ADIC3A	I	JS
					ADID3A	I	JS
					BNTG	M	JS
					FNTG	M	JS
					PROPB	I	JS
					PROPIN	I	JS
					STP3	I	JS
					TOL3	I	JS
KOP		Counts number of times constraint misses are halved down because of divergance problems	/XCODES/(	154)	TEST	M	KOP
					TOPM	I	KOP
KPST		Controls logic for compute interval during adjoint integration	/XCODES/(	155)	BNTG	M	KPST
					FNTG	M	KPST
K		Storage retrieval flag indicates end of arc, phase, or data	/XCODES/(	156)	AST3	O	K
					FNTG	O	K
					GETIT	M	K
					MODELA	I	K
					SDINP	M	K
KST		Arc or phase cut-off flag	/XCODES/(	157)	ADJUST	I	KST
					FNTG	M	KST
NCASE		Case number	/XCODES/(	159)	TOPM	O	NCASE
NCN		Number of elements in dΨ	/XCODES/(	160)	ADEQ3A	I	NCN
					ADICB3	I	NCN
					ADIC3A	I	NCN
					ADID3A	I	NCN
					ADJUST	I	NCN
					AST3	M	NCN
					BNTG	I	NCN
					BSTO3	I	NCN
					MTX3A	I	NCN
					OUT	I	NCN
					PAYO2	M	NCN
					TEST	M	NCN
					TOPM	I	NCN
					TRAN3	I	NCN
					TRTOSZ	I	NCN
NEQB		Number of integrated quantities during adjoint solution	/XCODES/(	161)	ADICB3	I	NEQB
					ADIC3A	M	NEQB
					RKTB3A	I	NN

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NEQ		Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
					ADIC3A	I	NEQ
					ADIO3A	I	NEQ
					AGETB3	I	NEQ
					AST3	I	NEQ
					BGET3	I	NEQ
					BSTO3	I	NEQ
					MTX3A	I	NEQ
					OUT	I	NEQ
					REU3	I	NEQ
					SDER3	I	NEQ
					SDINP	M	NEQ
					TOPM	I	NEQ
					TRAN3	I	NEQ
					YREF3	I	NEQ
NOP		Counts number of times payoff is scaled down due to divergance problems	/XCODES/(	163)	TEST	M	NOP
					TOPM	I	NOP
NPH		Number of phases in trajectory	/XCODES/(	164)	BNTG	I	NPH
					FNTG	O	NPH
					PRMSET	I	NPH
					SDINP	M	NPH
					TEST	I	NPH
					TOPM	I	NPH
N		Description not input	/XCODES/(	165)	MTX3A	M	N
					PAYO2	M	N
					SDINP	M	N
NST		Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
					FNTG	O	NST
					PROPB	I	NST
					SDINP	I	NS
					SDINP	M	NST
					TEST	I	NST
					TOPM	I	NST
					TRAN3	I	NST
IPST		Phase counter for first nominal trajectory	/XCODES/(	167)	AST3	I	IPST
					FNTG	M	IPST
					GUI3A	I	IPST
					REU3	I	IPST
IPRINT		Print page counter initialization flag	/XCODES/(	168)	OUT	M	IPRINT
					TEST	I	IPRINT
					TOPM	O	IPRINT
ISTN		Stored history data arc number	/XCODES/(	169)	AGETB3	O	ISTN
					AST3	O	ISTN
					GETIT	I	ISTN
					TRAN3	I	ISTN
IPHN		Stored history data phase number	/XCODES/(	170)	AGETB3	M	IPHN
					AST3	M	IPHN
					GETIT	I	IPHN
IBLK1		Storage retrieval buffer counter	/XCODES/(	173)	AST3	M	IBLK1
					BSTO3	M	IBLK1
IBLK2		Storage retrieval buffer counter	/XCODES/(	174)	AGETB3	M	IBLK2
					AST3	M	IBLK2
ISTOP		Arc cut-off flag	/XCODES/(	175)	FNTG	M	ISTOP
					STP3	I	ISTOP
ISTPP		Phase cut-off flag	/XCODES/(	176)	FNTG	M	ISTPP
					STP3	I	ISTPP
L		Integration traffic control flag	/XCODES/(	177)	BNTG	M	L
		L = 1 means evaluate derivatives			FNTG	M	L
		= 2 check cut-off			OUT	I	L
		= 3 print or cut-off detected			RKTA3A	M	L
					RKTB3A	M	L
					SDINP	M	L

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IFOB		Forward or adjoint integration flag = 1 means forward = 2 means adjoint	/XCODES/(	178)	ACCEL	I	IFOB
					BEROCO	I	IFOB
					BLYNE	I	IFOB
					EQUA3	I	IFOB
					IMPUL	I	IFOB
					SPLYNE	I	IFOB
					TOPM	O	IFOB
NB		Extent of integration set during adjoints on branch problem	/XCODES/(	179)	ADEQ3A	I	NB
					ADICB3	M	NB
					ADIC3A	M	NB
					BNTG	O	NB
					RKTB3A	I	NB
					STVRL3	I	NB
LB		First element number in partitioned dY vector	/XCODES/(	180)	ADEQ3A	I	LB
					ADICB3	O	LB
					ADIC3A	M	NN
MB		Last element number in partitioned dY vector	/XCODES/(	181)	ADEQ3A	I	MB
					ADICB3	O	MB
					ADIC3A	O	MB
NPHP		Description not input	/XCODES/(	182)	REU3	O	NPHP
					TEST	I	NPHP
NPHB		Description not input	/XCODES/(	183)	REU3	O	NPHB
					TEST	I	NPHB
NCTIN		Number of elements in upper triangular portion of A matrix	/XCODES/(	184)	ADICB3	I	NCTIN
					ADIC3A	M	NCTIN
NEQF		Number of equations to be integrated on forward trajectory	/XCODES/(	185)	REU3	I	NEQF
					RKTA3A	I	NN
					SDINP	O	NEQF
					STAU	I	NEQF
					TOPM	O	NEQF
					TRAN3	O	NEQF
JIN		Description not input	/XCODES/(	186)	EQUA3	I	JIN
					MODELA	M	JIN
					MODELB	O	ILAB
					PROPIN	O	ILAB
INEDS		Description not input	/XCODES/(	187)	BNTG	I	INEDS
					PROPB	O	INEDS
JPRP		Propulsion flag for different rocket options	/XCODES/(	194)	ACCEL	I	JPRP
					DER3A	I	JPRP
					EQUA3	I	JPRP
					MODELA	I	JPRP
					PDY3A	I	JPRP
					PROPB	O	JPRP
					PROPIN	O	JPRP
JGII		Control option	/XCODES/(	195)	ACCEL	I	JGII
					BNTG	O	JGII
					DER3A	I	JGII
					FNTG	M	JGII
					GUI3A	I	JGII
					MODELA	I	JGII
					MODELB	I	JGII
					MTX3A	I	JGII
					PDY3A	I	JGII
MTT		Thrust curve number	/XCODES/(	196)	EQUA3	I	MTT
					PROPB	M	MTT
					PROPIN	M	MTT
MPIN		Save thrust curve numbers for adjoint solution	/XCODES/(	197)	PROPB	I	MPIN
					PROPIN	O	MPIN
JP1		Option flag for first governing equation	/XCODES/(	217)	AGETB3	M	JP1
					AST3	M	JP1
					MODELA	M	JP1
					MODELB	I	JP1
					PROPB	O	JP1
					PROPIN	O	JP1

FORTRAN SYMBOL	MATH SYMBOL	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
JP2		Option flag for second governing equation	/XCODES/(	218)	MODELA	I	JP2
					MODELB	I	JP2
					PROPB	O	JP2
					PROPIN	O	JP2
JP3		Option flag for third governing equation	/XCODES/(	219)	AGETB3	O	JP3
					AST3	M	JP3
					MODELA	M	JP3
					MODELB	I	JP3
					OUT	I	JP3
					PROPIN	O	JP3

SUBRØUTINE ACCEL

Subroutine ACCEL

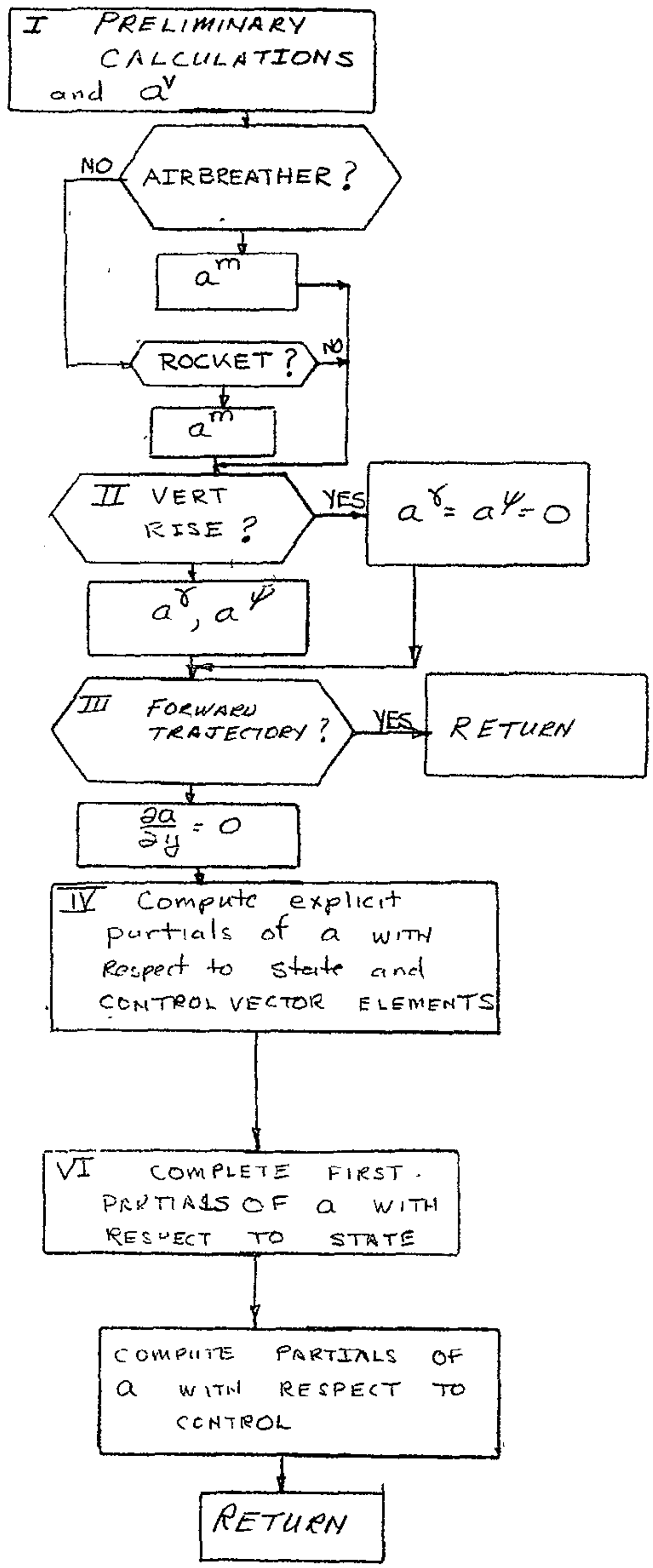
Purpose

Computes acceleration vector, a, and its partials with respect to state and elements of the control vector. It also computes the complete first partials of the acceleration vector with respect to the state. See Section 9.3 of Volume I.

Description

During forward trajectory, a^v, a^γ, a^ψ and a^m are computed. On the adjoint solution, matrix $\frac{\partial a}{\partial y}$ is also computed. The acceleration vector a is defined in equation (2.4-2) of Volume I.

SUBROUTINE ACCEL



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AE	A_{exit}	I	Total nozzle exit area	/GENF /(	520)	ACCEL FM2 IMPUL PROPB PROPIN SDER3	I I I O O I	AE AE AE AE AE AE
AG	a^{γ}	M	Acceleration vector element.	/AXL /(	2)	ACCEL DER3A PDY3A	M I I	AG AG AG
AGA		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	AGA
AGD		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	AGD
AGM		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	30)	ACCEL PDY3A	O I	AGM AGM
AGR		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	18)	ACCEL PDY3A	O I	AGR AGR
AGT		O	Element of AW matrix	/ACCEL /(*	)	ACCEL	O	AGT
AGV		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	6)	ACCEL PDY3A	O I	AGV AGV
AM	a^{m}	M	Acceleration vector element.	/AXL /(	4)	ACCEL DER3A	M I	AM AM
AMR		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	20)	ACCEL PDY3A	O I	AMR AMR
AMT		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	AMT
AMV		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	8)	ACCEL PDY3A	O I	AMV AMV
AP	a^{ψ}	M	Acceleration vector element.	/AXL /(	3)	ACCEL DER3A PDY3A	M I I	AP AP AP
APA		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	APA
APD		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	APD
APM		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	31)	ACCEL PDY3A	O I	APM APM
APR		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	19)	ACCEL PDY3A	O I	APR APR
APT		O	Element of AW matrix	/ACCEL /(*	)	ACCEL	O	APT
APV		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	7)	ACCEL PDY3A	O I	APV APV
AV	a^{v}	M	Acceleration vector element.	/AXL /(	1)	ACCEL DER3A	M I	AV AV
AVA		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	AVA
AVD		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	AVD
AVM		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	29)	ACCEL PDY3A	O I	AVM AVM
AVR		O	Element of matrix of acceleration vector partials WRT state.	/AXL /(	17)	ACCEL PDY3A	O I	AVR AVR
AVT		W	Element of AW matrix	/ACCEL /(*	)	ACCEL	W	AVT
AVV		M	Name of acceleration partials matrix.	/AXL /(	5)	ACCEL ACCEL PDY3A	M O I	AVV AY AVV
AW	$\partial a/\partial w$	O	Matrix of partials of acceleration vector WRT inplane control vector	/ACCEL /(*	)	ACCEL	O	AW
AY		O	Name of acceleration partials matrix.	/AXL /(	5)	ACCEL ACCEL PDY3A	M O I	AVV AY AVV

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
CODAE	$\cos(\alpha-\delta_E)$	I	See symbol	/GENF /(	549)	ACCEL	I	CODAE
						BL4	I	CODAE
						BL6	I	CODAE
						BL7	I	CODAE
						BL8	I	CODAE
						FH3	I	CODAE
						SDER3	I	CODAE
						VT	O	CODAE
COSA	$\cos\alpha$	I	See symbol	/AECO3 /(	8)	ACCEL	I	COSA
						BL4	I	COSA
						BL6	I	COSA
						BL7	I	COSA
						BL8	I	COSA
						FH3	I	COSA
						OUT	I	COSA
						VT	M	COSA
COSGAM	$\cos(\gamma)$	I	See symbol	/STATE3/(	687)	ACCEL	I	COSGAM
						BL4	I	COSGAM
						BL8	I	COSGAM
						DER3A	I	COSGAM
						EQUA3	O	COSGAM
						MODELA	I	COSGAM
						MODELB	I	COSGAM
						OUT	I	COSGAM
						PDBC	I	COSGAM
						PDY3A	I	COSGAM
COSPHI	$\cos\phi$	I	See symbol	/AECO3 /(	13)	ACCEL	I	COSPHI
						BL4	I	COSPHI
						GUI3A	M	COSPHI
						MODELA	M	COSPHI
						MODELB	M	COSPHI
						OUT	I	COSPHI
DB	D_b	I	Base drag (LBS)	/GENF /(	537)	ACCEL	I	DB
						BL4	I	DB
						BL6	I	DB
						BL7	I	DB
						BL8	I	DB
						EQUA3	I	DB
						FH3	I	DB
						OUT	I	DB
						SDER3	I	DB
						VT	I	DB
DBR		I	Partial of base drag wrt altitude	/GENF /(	536)	ACCEL	I	DBR
						BL4	I	DBR
						BL6	I	DBR
						BL7	I	DBR
						BL8	I	DBR
						EQUA3	I	DBR
						FH3	I	DBR
						VT	I	DBR
DPDY	$\partial\omega/\partial y$	I	Matrix of partials of in-plane control vector wrt state	/GENF /(	610)	ACCEL	I	DPDY
						BLGCON	I	DPDY
						OUT	I	DPDY
DRAG	D	I	Aerodynamic drag (LBS)	/GENF /(	497)	ACCEL	I	DRAG
						BL5	I	DRAG
						BL7	I	DRAG
						BL8	I	DRAG
						ENVPRM	I	DRAG
						FH3	I	DRAG
						OUT	I	DRAG
						PROPB	O	DRAG
						PROPIN	O	DRAG
						SDER3	I	DRAG
						VT	M	DRAG

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DRAGA		I	Partial of drag wrt angle of attack	/GENF /(	534)	ACCEL	I	DRAGA
						BL5	I	DRAGA
						BL7	I	DRAGA
						BL8	I	DRAGA
						FH3	I	DRAGA
						VT	M	DRAGA
DRAGR		I	Partial drag wrt altitude	/GENF /(	533)	ACCEL	I	DRAGR
						BL5	I	DRAGR
						BL7	I	DRAGR
						BL8	I	DRAGR
						FH3	I	DRAGR
						VT	M	DRAGR
DRAGV		I	Partial of drag wrt velocity	/GENF /(	532)	ACCEL	I	DRAGV
						BL5	I	DRAGV
						BL7	I	DRAGV
						BL8	I	DRAGV
						FH3	I	DRAGV
						VT	M	DRAGV
FVAC		I	Total vacuum thrust [rocket] (LBS)	/GENF /(	528)	ACCEL	I	FVAC
						EQUA3	M	FVAC
						FH2	I	FVAC
						IMPUL	M	FVAC
						PROPB	M	FVAC
						PROPIN	M	FVAC
						SDER3	I	FVAC
GDA	$\partial\dot{\gamma}/\partial\alpha$	O	See symbol	/AECO3 /(	5)	ACCEL	O	GDA
						ADEQ3A	I	GDA
GDPH	$\partial\dot{\gamma}/\partial\phi$	O	See symbol	/AECO3 /(	14)	ACCEL	O	GDPH
						ADEQ3A	I	GDPH
						PDY3A	O	GDPH
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
IFOB		I	Forward or adjoint integration flag = 1 means forward = 2 means adjoint	/XCODES/(	178)	ACCEL	I	IFOB
						BEROCO	I	IFOB
						BLYNE	I	IFOB
						EQUA3	I	IFOB
						IMPUL	I	IFOB
						SPLYNE	I	IFOB
						TOPM	O	IFOB
ISP	I_{sp}	I	Net vacuum specific impulse	/GENF /(	538)	ACCEL	I	ISP
						IMPUL	O	ISP
ISPF		I	Partial of ISP wrt vacuum thrust	/GENF /(	539)	ACCEL	I	ISPF
						IMPUL	O	ISPF

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
JGII		I	Control option	/XCODES/(	195)	ACCEL	I	JGII
						BNTG	O	JGII
						DER3A	I	JGII
						FNTG	M	JGII
						GUI3A	I	JGII
						MODELA	I	JGII
						MODELB	I	JGII
						MTX3A	I	JGII
						PDY3A	I	JGII
JPRP		I	Propulsion flag for different rocket options	/XCODES/(	194)	ACCEL	I	JPRP
						DER3A	I	JPRP
						EQUA3	I	JPRP
						MODELA	I	JPRP
						PDY3A	I	JPRP
						PROPB	O	JPRP
						PROPIN	O	JPRP
LIFT	L	I	Aerodynamic lift (LBS)	/GENF /(	496)	ACCEL	I	LIFT
						BL4	I	LIFT
						BL5	I	LIFT
						BL6	I	LIFT
						ENVPRM	I	LIFT
						FH3	I	LIFT
						OUT	I	LIFT
						PROPB	O	LIFT
						PROPIN	O	LIFT
						VT	O	LIFT
LIFTA		I	Partial of lift wrt angle-of-attack	/GENF /(	531)	ACCEL	I	LIFTA
						BL4	I	LIFTA
						BL5	I	LIFTA
						BL6	I	LIFTA
						FH3	I	LIFTA
						VT	O	LIFTA
LIFTM		I	Partial of LIFT wrt mass	/GENF /(	535)	ACCEL	I	LIFTM
						BL4	I	LIFTM
						BL5	I	LIFTM
						BL6	I	LIFTM
						FH3	I	LIFTM
						VT	O	LIFTM
LIFTR		I	Partial of lift wrt altitude	/GENF /(	530)	ACCEL	I	LIFTR
						BL4	I	LIFTR
						BL5	I	LIFTR
						BL6	I	LIFTR
						FH3	I	LIFTR
						VT	O	LIFTR
LIFTV		I	Partial of lift wrt velocity	/GENF /(	529)	ACCEL	I	LIFTV
						BL4	I	LIFTV
						BL5	I	LIFTV
						BL6	I	LIFTV
						FH3	I	LIFTV
						VT	O	LIFTV
M	m	I	Mass	/STATE3/(	4)	ACCEL	I	M
						BL4	I	M
						BL8	I	M
						EQUA3	I	M
						OUT	I	M
						SDER3	I	M
PAR		I	Deriv Of press Wrt alt.	/GENF /(	312)	ACCEL	I	PAR
						FH2	I	PAR
PDA	$\partial\dot{\psi}/\partial\alpha$	O	See symbol	/AECO3 /(	6)	ACCEL	O	PDA
						ADEQ3A	I	PDA
PDPH	$\partial\dot{\psi}/\partial\phi$	O	See symbol	/AECO3 /(	15)	ACCEL	O	PDPH
						ADEQ3A	I	PDPH
						PDY3A	O	PDPH
SFC		I	Specific fuel consumption (LBS/LB/HR)	/AIRBRE/(	4)	ACCEL	I	SFC
SFCH		I	Partial of SFC WRT altitude.	/AIRBRE/(	6)	ACCEL	I	SFCH
SFCV		I	Partial of SFC WRT velocity.	/AIRBRE/(	5)	ACCEL	I	SFCV

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SIDAE	$\sin(\alpha-\delta_E)$	I	See symbol		/GENF /(	557)	ACCEL	I	SIDAE
							BL4	I	SIDAE
							BL6	I	SIDAE
							BL7	I	SIDAE
							BL8	I	SIDAE
							FH3	I	SIDAE
							VT	O	SIDAE
SINA	$\sin\alpha$	I	See symbol		/AECO3 /(	7)	ACCEL	I	SINA
							BL4	I	SINA
							BL6	I	SINA
							BL7	I	SINA
							BL8	I	SINA
							FH3	I	SINA
							GUI3A	M	SINA
							OUT	I	SINA
							VT	M	SINA
SINPHI	$\sin\phi$	I	See symbol		/AECO3 /(	12)	ACCEL	I	SINPHI
							BL4	I	SINPHI
							MODELA	M	SINPHI
							MODELB	M	SINPHI
							OUT	I	SINPHI
T	T	I	Thrust	(LBS)	/GENF /(	411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T
TAIRBH		I	Partial of airbreather thrust WRT altitude		/AIRBRE/(	3)	ACCEL	I	TAIRBH
							FH4	I	TAIRBH
TAIRBV		I	Partial of airbreather thrust WRT velocity		/AIRBRE/(	2)	ACCEL	I	TAIRBV
							FH4	I	TAIRBV
V	v	I	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	V
							STP3	I	VAR
							TOPM	O	KWOW
							VREF3	M	V

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
VDA	$\partial\dot{v}/\partial\alpha$	O	See symbol	/AEC03 /(	4)	ACCEL ADEQ3A	O I	VDA VDA
XK2A		I	Partial of governing equation wrt state or control vector component	/GENF /(	582)	ACCEL BLGCON EL2	I I O	XK2A XK2A XK2A
XK2D		I	Partial of governing equation wrt state or control vector component	/GENF /(	579)	ACCEL BLGCON EL1 EL2	I I O O	XK2D XK2D XK2D XK2D

30 OCT '72 G 01-46

```
      SUBROUTINE ACCEL                                                  ACCEL
C                                                                       COMN
C           COMPUTES   ACCELERATION VECTOR  AND ITS  PARTIALS           COMN
C           WITH RESPECT TO  STATE                                      COMN
C                                                                       COMN
      COMMON/GLOBAL/                                                    GLOBAL
     *GR         ,ER         ,CMGZ       ,XLAMRF     ,YMURF      ,LUM   GLOBAL
     *,JJOP(10)  ,IFATAL     ,NARC       ,NBRAN      ,NFARC      ,ID(4) GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG   ,MAXTAB                                GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)     GLOBAL
     *,ITPSO ,KSOL     ,KGLOBL(8)                                       RETAP
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)     ,SVAR(10)   ,WDC(20)    ,  GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)    ,DTP       ,  GENF
     *DTS        ,DT         ,G          ,DPSQ       ,Q          ,QS        ,  GENF
     *R          ,RE         ,MACH       ,PA         ,RO         ,CS        ,  GENF
     *VNU        ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ     ,  GENF
     *SVSQ       ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)     ,  GENF
     *TST(20)    ,TPH    (20),DIS(20)    ,DIP(20)    ,T          ,W         ,  GENF
     *TLP1(20)   ,TLS1   (20),DIP1(20)   ,DIS1(20)   ,TIME       ,OMP       ,  GENF
     *TIMPR      ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)   ,  GENF
     *AE         ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV     ,  GENF
     *QR         ,QV         ,FVAC       ,LIFTV      ,                        GENF
     *LIFTR      ,LIFTA      ,           DRAGV      ,DRAGR      ,DRAGA     ,  GENF
     *           LIFTM      ,DBR        ,DB         ,ISP        ,ISPF      ,  GENF
     *                      ULFT       ,ULFTV      ,ULFTR      ,ULFTA     ,  GENF
     *XMCG       ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE     ,  GENF
     *CULFT      ,CT         ,CALPHA     ,COE        ,DELTAE     ,SID       ,  GENF
     *COD        ,SIDAE      ,XCG        ,ZCG        ,XJ                      GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP       ,  GENF
     *FRATED   ,IRATED       ,                                          GENF
     *P1         ,P2         ,P3       ,XK1          ,XK2        ,XK3       ,  GENF
     *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D      ,  GENF
     *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V      ,  GENF
     *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P      ,  GENF
     *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O      ,  GENF
     *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M      ,  GENF
     *PV         ,PG         ,PP         ,PR         ,PO         ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/AECO3/                                                     AECO3
     *APHO       ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA       ,  AECO3
     *SINA       ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI    ,  AECO3
     *COSPHI     ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDO       ,  AECO3
     *CDOM       ,CLO        ,FK         ,XCGM       ,ZCGM       ,CLOM      ,  AECO3
     *CM         ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM       ,FKM       ,  AECO3
     *CLAM       ,CL         ,CLA        ,CLM        ,                        AECO3
     *CD         ,CDA        ,CDM                                       AECO3
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR   (20),ITI        ,INTB       ,JGID(20,2),JPH (20,2),  XCODES
     *JST    (20)            ,NCNST      ,NSB        ,NSAB       ,NICNB     ,  XCODES
     *IZOP       ,ICOP       ,IFAW       ,IFAR       ,IFB        ,IND       ,  XCODES
     *IOPEN      ,IPH        ,ISPH       ,ISST       ,IARC       ,ISTART    ,  XCODES
     *ITCT       ,ITER       ,IVAR       ,JK         ,JPS        ,JS        ,  XCODES
     *KOP        ,KPST       ,K          ,KST        ,NAD        ,NCASE     ,  XCODES
     *NCN        ,NEQB       ,NEQ        ,NOP        ,NPH        ,N         ,  XCODES
     *NST        ,IPST       ,IPRINT     ,ISTN       ,IPHN       ,ISTNB     ,  XCODES
     *IPHNB      ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP      ,L         ,  XCODES
     *IFOB       ,NB         ,LB         ,MB         ,NPHP       ,NPHB      ,  XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99)  ,DVARL(99)  ,VO(9)      ,SVY(10)   ,  STATE3D
     *XL(9,9)    ,YOP(20,9),YDS (20,9),COSGAM      ,SINGAM     ,SAVBP(15) ,  STATE3D
     *SINPSI     ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2    ,  STATE3D
     *SVBV    (9),OMEGA      ,OMEGA2     ,                                  STATE3D
     *VDV        ,GDV        ,RDV        ,MDV        ,PDV        ,ODV       ,  STATE3D
     *UDV        ,VDG        ,GDG        ,RDG        ,PDG        ,ODG       ,  STATE3D
     *UDG        ,VDR        ,GDR        ,MDR        ,PDR        ,ODR       ,  STATE3D
     *UDR        ,VDM        ,GDM        ,MDM        ,PDM        ,VDP       ,  STATE3D
     *GOP        ,PDP        ,ODP        ,UDP        ,VDO        ,GDO       ,  STATE3D
     *PDO        ,UDO        ,,HTDV      ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
```

```
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                    STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,  EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2), EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD), EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)            EQUV3
      REAL M,MU,MD                                                      EQUV3
      COMMON/ AXL/                                                      AXL
     *AV    ,AG    ,AP    ,AM    ,AVV   ,AGV   ,APV   ,AMV   ,AVG   ,AGG   ,APG   , AXL
     *AMG   ,AVP   ,AGP   ,APP   ,AMP   ,AVR   ,AGR   ,APR   ,AMR   ,AVO   ,AGO   , AXL
     *APO   ,AMO   ,AVU   ,AGU   ,APU   ,AMU   ,AVM   ,AGM   ,APM   ,AMM       AXL
      DIMENSION  AY(1)                                                  AXL
      EQUIVALENCE (AY,AVV)                                              AXL
      DIMENSION AW(12),AYIMP(28)                                        ACCEL
      EQUIVALENCE (AVT,AW(1)),(AVD,AW(5)),(AVA,AW(9)),(AGT,AW(2)),      ACCEL
     * (AGD,AW(6)),(AGA,AW(10)),(APT,AW(3)),(APD,AW(7)),(APA,AW(11)),   ACCEL
     *(AMT,AW(4))                                                       ACCEL
      DATA AW/12*0./                                                    ACCEL
      COMMON/AIRBRE/ TAIRB, TAIRBV,TAIRBH,SFC,SFCV,SFCH                 FIXED
C       I    PRELIMINARY  COMPUTATION                                   COMN
      TCDAE = T*CODAE                                                   ACCEL
      TSDAE = T*SIDAE                                                   ACCEL
      DBCA = DB*COSA                                                    ACCEL
      DBSA = DB*SINA                                                    ACCEL
      GEISP = GR*ISP                                                    ACCEL
      AV = (TCDAE - DBCA - DRAG)/M                                      ACCEL
      AM=0.                                                             P014
C       AIRBREATHER                                                     FIXED
      IF(JPRO.NE.2) GO TO 10                                            FIXED
      AM=-SFC*T/3600./GR                                                JULY28
      GO  TO 12                                                         FIXED
   10 CONTINUE                                                          FIXED
      IF(JPRP.GT 1) AM=-FVAC/GEISP                                      P014
   12 CONTINUE                                                          FIXED
      IF(JGII EQ 8) GO TO 40                                            ACCEL
C       II    TEST FOR VERTICAL RISE OR PITCHOVER                       COMN
      AX = (TSDAE - DBSA + LIFT)/M                                      ACCEL
      AG = AX*COSPHI                                                    ACCEL
      AP = AX*SINPHI                                                    ACCEL
      GO TO 45                                                          ACCEL
   40 AG=0.                                                             ACCEL
      AP=0.                                                             ACCEL
C       III   IF FORWARD TRAJECTORY RETURN                              COMN
   45 IF(IFOB.EQ.1) RETURN                                              ACCEL
      DO 46 I=1,28                                                      ACCEL
   46 AY(I)= 0.                                                         ACCEL
      DBRCA = DBR*COSA                                                  ACCEL
      DBRSA = DBR*SINA                                                  ACCEL
C       IV       ***        ***        ***                              COMN
C     COMPUTE THE EXPLICIT FIRST PARTIALS OF THE A-VECTOR WITH RESPECT  ACCEL
C     TO THE STATE AND DECISION VECTORS.                                ACCEL
   51 AVV = -DRAGV/M                                                    ACCEL
      AVR = (-DBRCA - DRAGR)/M                                          ACCEL
      AVM = -AV/M                                                       ACCEL
      IF(AM.EQ 0.) GO TO 53                                             P014
      IF(JPRO.NE.2) GO TO  52                                           FIXED
      AMT = -SFC/3600./GR                                               JULY28
      AMR = AMT*TAIRBH - T*SFCH/GR/3600.                                JULY28
      AMV = AMT*TAIRBV - T*SFCV/GR/3600.                                JULY28
      GO  TO  54                                                        FIXED
   52 CONTINUE                                                          FIXED
      AMT =-(1. - ISPF*FVAC/ISP)/GEISP                                  ACCEL
      AMR = AE*PAR*AMT                                                  ACCEL
      GO TO 54                                                          P014
   53 AMT=0.                                                            P014
      AMR=0.                                                            P014
   54 CONTINUE                                                          P014
      AVT = CDDAE/M                                                     ACCEL
      AVD = TSDAE/M                                                     ACCEL
      AVA = (-TSDAE + DBSA - DRAGA)/M                                   ACCEL
C       V     SKIP  ON  VERTICAL RISE OR PITCHOVER                      COMN
      IF(JGII EQ.8) GO TO 103                                           ACCEL
```

```
      AXV = LIFTV/M                                                     ACCEL
      AXR = (-DBRSA + LIFTR)/M                                          ACCEL
      AXM = (LIFTM - AX)/M                                              ACCEL
      AGV = AXV*COSPHI                                                  ACCEL
      AGR = AXR*COSPHI                                                  ACCEL
      AGM = AXM*COSPHI                                                  ACCEL
      APV = AXV*SINPHI                                                  ACCEL
      APR = AXR*SINPHI                                                  ACCEL
      APM = AXM*SINPHI                                                  ACCEL
      AXT = SIDAE/M                                                     ACCEL
      AXD =-TCDAE/M                                                     ACCEL
      AXA = (TCDAE - DBCA + LIFTA)/M                                    ACCEL
      AGT = AXT*COSPHI                                                  ACCEL
      AGD = AXD*COSPHI                                                  ACCEL
      AGA = AXA*COSPHI                                                  ACCEL
      APT = AXT*SINPHI                                                  ACCEL
      APD = AXD*SINPHI                                                  ACCEL
      APA = AXA*SINPHI                                                  ACCEL
C       VI        ***                ***                                COMN
C     FIRST PARTIALS OF THE A-VECTOR WITH RESPECT TO THE STATE.         ACCEL
  103 CALL MATMLT(AYIMP,AVT,DPDY,4,3,7)                                 ACCEL
C       VII       ***                                                   COMN
C     COMPUTE THE TOTAL FIRST PARTIALS OF THE A-VECTOR WITH RESPECT TO  ACCEL
C     THE STATE.                                                        ACCEL
  104 CALL MATADD(AVV,AVV,AYIMP,4, 7)                                   ACCEL
C         PARTIALS  WITH  RESPECT  TO  CONTROL                          COMN
      VDA=AVA-AVD*XK2A/XK2D                                             ACCEL
      GDA=(AGA-AGD*XK2A/XK2D)/ V                                        ACCEL
      PDA=(APA-APD*XK2A/XK2D)/(V*COSGAM)                                ACCEL
      GDPH=-AP/V                                                        ACCEL
      PDPH=AG/(V*COSGAM)                                                ACCEL
      RETURN                                                            ACCEL
      END                                                               ACCEL
```

20 OCT 72 6.01-46

SUBRØUTINE ADEQ3A

Subroutine ADEQ3A

Entry Point. ADEQ

Purpose

ADEQ3A computes the derivatives of the adjoint differential equations as expressed in equations (12.3-3) through (12.3-8) in Volume I. It also supplies the integrands of the upper triangular portion of the A matrix as given in equation (12.1-20). These, of course, contain functions of the impulse response function as defined in equation (11.1-16).

Description

This routine is called from BNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	A	O	Control Integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BSTO3	I	A
						MTX3A	I	A
						PAYO2	I	A
						SDINP	I	A
						TRAN3	I	A
DVARL		O	Array of derivatives for adjoint integration	/STATE3/(	128)	ADEQ3A	O	DVARL
						ADICB3	O	DVARL
						ADIC3A	O	DVARL
						RKTB3A	I	DY
GDA	$\partial\dot{\gamma}/\partial\alpha$	I	See symbol	/AECO3 /(	5)	ACCEL	O	GDA
						ADEQ3A	I	GDA
GDG		I	Partial derivative of equation of motion WRT state	/STATE3/(	729)	ADEQ3A	I	GDG
						PDY3A	O	GDG
GDM		I	Partial derivative of equation of motion WRT state	/STATE3/(	741)	ADEQ3A	I	GDM
						PDY3A	O	GDM
GDO		I	Partial derivative of equation of motion WRT state	/STATE3/(	750)	ADEQ3A	I	GDO
						PDY3A	O	GDO
GDP		I	Partial derivative of equation of motion WRT state	/STATE3/(	745)	ADEQ3A	I	GDP
						PDY3A	O	GDP
GDPH	$\partial\dot{\gamma}/\partial\phi$	I	See symbol	/AECO3 /(	14)	ACCEL	O	GDPH
						ADEQ3A	I	GDPH
						PDY3A	O	GDPH
GDR		I	Partial derivative of equation of motion WRT state	/STATE3/(	735)	ADEQ3A	I	GDR
						PDY3A	O	GDR
GDV		I	Partial derivative of equation of motion WRT state	/STATE3/(	722)	ADEQ3A	I	GDV
						PDY3A	O	GDV
HTDR		I	Partial derivative of equation of motion WRT state	/STATE3/(	754)	ADEQ3A	I	HTDR
						PDY3A	O	HTDR
HTDV		I	Partial derivative of equation of motion WRT state	/STATE3/(	753)	ADEQ3A	I	HTDV
						PDY3A	O	HTDV
LB		I	First element number in partitioned d¥ vector	/XCODES/(	180)	ADEQ3A	I	LB
						ADICB3	O	LB
						ADIC3A	M	NN
MB		I	Last element number in partitioned d¥ vector	/XCODES/(	181)	ADEQ3A	I	MB
						ADICB3	O	MB
						ADIC3A	O	MB
MDM		I	Partial derivative of equation of motion WRT state	/STATE3/(	742)	ADEQ3A	I	MDM
						PDY3A	O	MDM
MDR		I	Partial derivative of equation of motion WRT state	/STATE3/(	736)	ADEQ3A	I	MDR
						PDY3A	O	MDR
MDV		I	Partial derivative of equation of motion WRT state	/STATE3/(	724)	ADEQ3A	I	MDV
						PDY3A	O	MDV
NB		I	Extent of integration set during adjoints on branch problem	/XCODES/(	179)	ADEQ3A	I	NB
						ADICB3	M	NB
						ADIC3A	M	NB
						BNTG	O	NB
						RKTB3A	I	NB
						STVRL3	I	NB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NCN		I	Number of elements in $d\dot{\Psi}$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BSTO3	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAYO2	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
ODG		I	Partial derivative of equation of motion WRT state	/STATE3/(	732)	ADEQ3A	I	ODG
						PDY3A	O	ODG
ODP		I	Partial derivative of equation of motion WRT state	/STATE3/(	747)	ADEQ3A	I	ODP
						PDY3A	O	ODP
ODR		I	Partial derivative of equation of motion WRT state	/STATE3/(	738)	ADEQ3A	I	ODR
						PDY3A	O	ODR
ODV		I	Partial derivative of equation of motion WRT state	/STATE3/(	726)	ADEQ3A	I	ODV
						PDY3A	M	ODV
PDA	$\partial\dot{\Psi}/\partial\alpha$	I	See symbol	/AECO3 /(	6)	ACCEL	O	PDA
						ADEQ3A	I	PDA
PDG		I	Partial derivative of equation of motion WRT state	/STATE3/(	731)	ADEQ3A	I	PDG
						PDY3A	O	PDG
PDM		I	Partial derivative of equation of motion WRT state	/STATE3/(	743)	ADEQ3A	I	PDM
						PDY3A	O	PDM
PDO		I	Partial derivative of equation of motion WRT state	/STATE3/(	751)	ADEQ3A	I	PDO
						PDY3A	O	PDO
PDP		I	Partial derivative of equation of motion WRT state	/STATE3/(	746)	ADEQ3A	I	PDP
						PDY3A	O	PDP
PDPH	$\partial\dot{\Psi}/\partial\phi$	I	See symbol	/AECO3 /(	15)	ACCEL	O	PDPH
						ADEQ3A	I	PDPH
						PDY3A	O	PDPH
PDR		I	Partial derivative of equation of motion WRT state	/STATE3/(	737)	ADEQ3A	I	PDR
						PDY3A	O	PDR
PDV		I	Partial derivative of equation of motion WRT state	/STATE3/(	725)	ADEQ3A	I	PDV
						PDY3A	O	PDV
RDG		I	Partial derivative of equation of motion WRT state	/STATE3/(	730)	ADEQ3A	I	RDG
						PDY3A	O	RDG
RDV		I	Partial derivative of equation of motion WRT state	/STATE3/(	723)	ADEQ3A	I	RDV
						PDY3A	O	RDV
UDG		I	Partial derivative of equation of motion WRT state	/STATE3/(	733)	ADEQ3A	I	UDG
						PDY3A	O	UDG
UDO		I	Partial derivative of equation of motion WRT state	/STATE3/(	752)	ADEQ3A	I	UDO
						PDY3A	O	UDO
UDP		I	Partial derivative of equation of motion WRT state	/STATE3/(	748)	ADEQ3A	I	UDP
						PDY3A	O	UDP
UDR		I	Partial derivative of equation of motion WRT state	/STATE3/(	739)	ADEQ3A	I	UDR
						PDY3A	M	UDR
UDV		I	Partial derivative of equation of motion WRT state	/STATE3/(	727)	ADEQ3A	I	UDV
						PDY3A	M	UDV
VARL		I	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A	I	VARL
						ADICB3	O	VARL
						ADIC3A	O	VARL
						PROPIN	I	ZZ
						RKTA3A	M	F
						RKTB3A	M	Y
						STVRL3	O	VARL
						TRAN3	M	VARL

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
VDA	$\partial\dot{v}/\partial\alpha$	I	See symbol	/AECO3 /(	4)	ACCEL	O	VDA
						ADEQ3A	I	VDA
VDG		I	Partial derivative of equation of motion WRT state	/STATE3/(	728)	ADEQ3A	I	VDG
						PDY3A	O	VDG
VDM		I	Partial derivative of equation of motion WRT state	/STATE3/(	740)	ADEQ3A	I	VDM
						PDY3A	O	VDM
VDO		I	Partial derivative of equation of motion WRT state	/STATE3/(	749)	ADEQ3A	I	VDO
						PDY3A	O	VDO
VDP		I	Partial derivative of equation of motion WRT state	/STATE3/(	744)	ADEQ3A	I	VDP
						PDY3A	O	VDP
VDR		I	Partial derivative of equation of motion WRT state	/STATE3/(	734)	ADEQ3A	I	VDR
						PDY3A	O	VDR
VDV		I	Partial derivative of equation of motion WRT state	/STATE3/(	721)	ADEQ3A	I	VDV
						PDY3A	O	VDV
WORK		I	Working array, contains TOPEN1, TOPEN2, and PHIWT	/STS /(	3)	ADEQ3A	I	WORK
						FNTG	I	WORK
						MODELB	O	WORK
						MTX3A	I	WORK
						SDINP	M	WORK
						TEST	M	WORK
XL	$\lambda^{\Psi_i,\Omega_j}$	M	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
XLAMA	$\Lambda^{\Psi_i,\Omega_j}$	M	Impulse response function column vector associated with angle of attack	/AECO3 /(	16)	ADEQ3A	M	XLAMA
						ADIC3A	O	XLAMA
						AST3	O	XLAMA
						BGET3	O	XLAMA
						BSTO3	M	XLAMA
						MTX3A	I	XLAMA
						TRAN3	M	XLAMA
XLAMP	$\Lambda^{\Psi_i,\Omega_j}$	M	Impulse response function column vector associated with bank angle	/AECO3 /(	25)	ADEQ3A	M	XLAMP
						ADIC3A	O	XLAMP
						AST3	O	XLAMP
						BGET3	O	XLAMP
						BSTO3	M	XLAMP
						MTX3A	I	XLAMP
						TRAN3	M	XLAMP

```
      SUBROUTINE ADEQ3A                                                 ADEQ3A
C           COMPUTE   DERIVATIVES OF ADJOINT  DIFFERENTIAL EQUATIONS    COMN
C           ALSO  IMPULSE  RESPONSE  FUNCTIONS                          COMN
C           AND  INTEGRANDS OF  CONTROL  INTEGRALS  (A MATRIX)          COMN
      COMMON/STS/                                                       STS
     *DPAY     ,PMIN     ,WORK (20),NWDS     ,IPC    (7),NITER     ,    STS
     *MNGA(20,2),MNGP(20,2),AR(200),IAD(20) ,INP(20),ISV(20)            STS
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)  ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)    ,SVY(10)   ,  STATE3D
     *XL(9,9)  ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM   ,SAVBF(15) ,   STATE3D
     *SINPSI   ,COSPSI   ,SINRHO   ,COSRHO   ,DCORHO   ,DCORO2   ,      STATE3D
     *SVBV   (9),OMEGA   ,OMEGA2   ,                                    STATE3D
     *VDV      ,GDV      ,RDV      ,MDV      ,PDV      ,ODV      ,      STATE3D
     *UDV      ,VDG      ,GDG      ,RDG      ,PDG      ,ODG      ,      STATE3D
     *UDG      ,VDR      ,GDR      ,MDR      ,PDR      ,ODR      ,      STATE3D
     *UDR      ,VDM      ,GDM      ,MDM      ,PDM      ,VDP      ,      STATE3D
     *GDP      ,PDP      ,ODP      ,UDP      ,VDO      ,GDO      ,      STATE3D
     *PDO      ,UDO      ,HTDV     ,HTDR                                STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO   ,COS2RO   ,COS2GM                                        STATE3D
      COMMON/AEC03/                                                     AEC03
     *APHO     ,APHR     ,ALPHA    ,VDA      ,GDA      ,PDA      ,      AEC03
     *SINA     ,COSA     ,PHIO     ,PHID     ,PHI      ,SINPHI   ,      AEC03
     *COSPHI   ,GDPH     ,PDPH     ,XLAMA(9) ,XLAMP(9) ,CDO      ,      AEC03
     *CDOM     ,CLO      ,FK       ,XCGM     ,ZCGM     ,CLOM     ,      AEC03
     *CM       ,CMA      ,CMAM ,CMM ,CMO     ,CMOM     ,FKM      ,      AEC03
     *CLAM     ,CL       ,CLA      ,CLM      ,                          AEC03
     *CD       ,CDA      ,CDM                                           AEC03
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI      ,INTB     ,JGIO(20,2),JFH (20,2),  XCODES
     *JST   (20)         ,NCNST    ,NSB      ,NSAB     ,NICNB    ,      XCODES
     *I2OP     ,ICOP     ,IFAW     ,IFAR     ,IFB      ,IND      ,      XCODES
     *IOPEN    ,IPH      ,ISPH     ,ISST     ,IARC     ,ISTART   ,      XCODES
     *ITCT     ,ITER     ,IVAR     ,JK       ,JPS      ,JS       ,      XCODES
     *KOP      ,KPST     ,K        ,KST      ,NAD      ,NCASE    ,      XCODES
     *NCN      ,NEQB     ,NEQ      ,NOP      ,NPH      ,N        ,      XCODES
     *NST      ,IPST     ,IPRINT   ,ISTN     ,IPHN     ,ISTNB    ,      XCODES
     *IPHNB    ,IBLK1    ,IBLK2    ,ISTOP    ,ISTPP    ,L        ,      XCODES
     *IFDB     ,NB       ,LB       ,MB       ,NPHP     ,NPHB     ,      XCODES
     *NCTIN    ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3        XCODES
      COMMON/GENF/                                                      GENF
     *OMG(20)  ,OMGP(20,2),VARQ(9)  ,TOL(9)   ,SVAR(10) ,WDC(20)  ,     GENF
     *A(9,9)   ,ACON(9)  ,BCON(9)  ,COT1(9,9) ,DCON(9)  ,OTP      ,     GENF
     *DTS      ,DT       ,G        ,DPSQ     ,Q        ,QS       ,      GENF
     *R        ,RE       ,MACH     ,PA       ,RO       ,CS       ,      GENF
     *VNU      ,PAR      ,ROR      ,CSR      ,VNR      ,SUMSQ    ,      GENF
     *SVSQ     ,TIMEPH   ,TIMES    ,TOP      ,TOS      ,TR(9)    ,      GENF
     *TST(20)  ,TPH   (20),DIS(20)  ,DIP(20)  ,T        ,W        ,     GENF
     *TLP1(20) ,TLS1  (20),DIP1(20) ,DIS1(20) ,TIME     ,OMP            GENF
     *TIMPR    ,LIFT     ,DRAG     ,TAX      ,TBURN    ,TBU(20)  ,      GENF
     *AE       ,FP       ,FPOLD    ,FPD      ,MACHR    ,MACHV    ,      GENF
     *QR       ,QV       ,FVAC     ,LIFTV    ,                          GENF
     *LIFTR    ,LIFTA    ,          DRAGV    ,DRAGR    ,DRAGA    ,      GENF
     *          LIFTM    ,DBR      ,DB       ,ISP      ,ISPF     ,      GENF
     *                    ULFT     ,ULFTV    ,ULFTR    ,ULFTA    ,      GENF
     *XMCG     ,XMCGV    ,XMCGR    ,XMCGA    ,XMCGM    ,CODAE    ,      GENF
     *CULFT    ,CT       ,CALPHA   ,CDE      ,DELTAE   ,SID      ,      GENF
     *COD      ,SIDAE    ,XCG      ,ZCG      ,XJ                        GENF
      COMMON / GENF  /                                                  GENF
     *XJV      ,XJR      ,GH       ,GAMMAD   ,XKG      ,XKP      ,      GENF
     *FRATED   ,IRATED   ,                                              GENF
     *P1       ,P2       ,P3      ,XK1       ,XK2      ,XK3      ,      GENF
     *XK1T     ,XK2T     ,XK3T     ,XK1D     ,XK2D     ,XK3D     ,      GENF
     *XK1A     ,XK2A     ,XK3A     ,XK1V     ,XK2V     ,XK3V     ,      GENF
     *XK1G     ,XK2G     ,XK3G     ,XK1P     ,XK2P     ,XK3P     ,      GENF
     *XK1R     ,XK2R     ,XK3R     ,XK1O     ,XK2O     ,XK3O     ,      GENF
     *XK1U     ,XK2U     ,XK3U     ,XK1M     ,XK2M     ,XK3M     ,      GENF
     *PV       ,PG       ,PP       ,PR       ,PO       ,DPDY(3,8)       GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      EQUIVALENCE (WORK(11),CMULT)                                      ADEQ3A
      ENTRY ADEQ                                                        ADEQ3A
```

```
C       I    SET RANGE OF INTEGRATION ARRAY                                 COMN
        JJ=NB                                                               ADEQ3A
        II=NB-1                                                             ADEQ3A
C        II   STORE ADJOINT DERIV. IN  INTEGRATION ARRAY                    COMN
        DO 20 I=LB,MB                                                       ADEQ3A
        DO 10 J=1,6                                                         ADEQ3A
        XL(J,I)=VARL(JJ)                                                    ADEQ3A
     10 JJ =JJ+1                                                            ADEQ3A
        DVARL(II+1)= -( XL(1,I)*VDV+ XL(2,I)*GDV+ XL(3,I)*RDV+XL(4,I)*      ADEQ3A
       1MDV+  XL(5,I)*PDV+ XL(6,I)*ODV+ XL(7,I)*UDV+ XL (8,I)*HTDV)         ADEQ3A
        DVARL(II+2)= -( XL(1,I)*VDG+ XL(2,I)*GDG+ XL(3,I)*RDG+XL(5,I)*   PD ADEQ3A
       1G+XL(6,I)*ODG+ XL(7,I)*UDG)                                         ADEQ3A
        DVARL(II+3)= -(XL(1,I)* VDR+ XL(2,I)*GDR+ XL(4,I)*MDR+XL(5,I)*    P ADEQ3A
       *DR + XL(6,I)*ODR + XL(7,I)*UDR +XL(8,I)*HTDR)                       PD14
        DVARL(II+4)= -(XL(1,I)* VDM+ XL(2,I)*GDM+XL(4,I)*MDM+XL(5,I)*PDM)   ADEQ3A
        DVARL(II+5) = -(XL(1,I)*VDP+ XL(2,I)*GDP+XL(5,I)*PDP+XL(6,I)*    OD ADEQ3A
       1P+ XL(7,I)*UDP)                                                     ADEQ3A
        DVARL(II+6) = -(XL(1,I)*VDO+ XL(2,I)*GDO+ XL(5,I)*PDO+XL(7,I)*   UD ADEQ3A
       1O)                                                                  ADEQ3A
C         III  COMPUTE  IMPULSE  RESPONSE  FUNCTIONS                        COMN
        XLAMA(I) =  -( XL(1,I) *VDA+XL(2,I)*GDA+XL(5,I)*PDA)                ADEQ3A
        XLAMP(I)=-(XL(2,I)*GDPH+XL(5,I)*PDPH)                               ADEQ3A
     20 II=II+6                                                             ADEQ3A
C         III-A A MATRIX  INTEGRANDS                                        COMN
        DO  40   MI=1,NCN                                                   ADEQ3A
        NI = MI                                                             ADEQ3A
         C=1.                                                               ADEQ3A
C        IV    SET BRANCH  CONSTRAINT  ELEMENTS TO ZERO AS APPROPRIATE      COMN
        IF(MI.LT LB.OR MI.GT.MB) C=0.                                       ADEQ3A
         XLAMA(MI)=XLAMA(MI)*C                                              ADEQ3A
        XLAMP(MI) = XLAMP(MI)*C                                             ADEQ3A
        DO   30 MJ=1,NI                                                     ADEQ3A
        II = II+1                                                           ADEQ3A
        A(MJ,MI) = VARL(II)*C                                               ADEQ3A
        DVARL(II) = XLAMA(MI) *XLAMA(MJ)* CMULT                             ADEQ3A
       1 + XLAMP(MI)* XLAMP(MJ)                                             ADEQ3A
       2  * WORK(10)                                                        ADEQ3A
     30 CONTINUE                                                            ADEQ3A
     40 CONTINUE                                                            ADEQ3A
        RETURN                                                              ADEQ3A
        END                                                                 ADEQ3A
```

20 OCT 72 G.01-46

SUBRØUTINE ADICB3

Subroutine ADICB3

Purpose

This subroutine performs calculations of adjoint discontinuities and initializations for branched and intermediate constraint problems. It contains 3 entry points, ADI3B, ADICB, and ADI2B. These entry points are explained below with cross reference to equations in Volume I and Roman numeral comments in the subroutine listing.

Description

I	Branch point approached from Branch 3. (Entry ADI3B)
I-A	Save A matrix at this corner point to permit restoration of complete matrix on trunk.
I-B	Reset time and state to end of branch. This will permit initiation of adjoint integration from end of branch 2.
II and III	Adjoint initial conditions are calculated to initiate Branch 2 and integration. The equation for the adjoint initial condition is (11 5-2).
IV	At intermediate arc constraint (ENTRY ADICB) This entry point sets up necessary data for initializing adjoints at end of intermediate constraint arc. Logic then proceeds to II and III where the initialization equation corresponding to (11.4-3) is computed.

The entry point in this routine is called from BNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	A	M	Control integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BSTO3	I	A
						MTX3A	I	A
						PAYO2	I	A
						SDINP	I	A
						TRAN3	I	A
COTI		M	Temp storage for a matrix also called B matrix	/GENF /(	208)	ADICB3	M	COTI
						MTX3A	M	B
						TRAN3	M	COTI
DFD	$\dot{\psi}$	W	Rate of change of constraint	/ADICB3/(*	)	ADICB3	W	DFD
DVAR	$\dot{y}$	M	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
DVARL		O	Array of derivatives for adjoint integration	/STATE3/(	128)	ADEQ3A	O	DVARL
						ADICB3	O	DVARL
						ADIC3A	O	DVARL
						RKTB3A	I	DY
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
ISKP		W	Flag to stop redundant computation in subroutine PDBC	/ADICB3/(*	)	ADICB3	W	ISKP
ITQ		I	Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
						ADIC3A	I	ITQ
						ADID3A	I	ITQ
						CON3	I	ITQ
						SDINP	M	ITQ
						STAU	I	ITQ
						TOPM	D	IITQ
JS		M	Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
						ADIC3A	I	JS
						ADID3A	I	JS
						BNTG	M	JS
						FNTG	M	JS
						PROPB	I	JS
						PROPIN	I	JS
						STP3	I	JS
						TOL3	I	JS

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
JST		I	Arc cut-off option flag	/XCODES/(	112)	ADICB3	I	JST
						BNTG	I	JST
						FNTG	I	JST
						SDINP	M	JST
LB		O	First element number in partitioned $d\Psi$ vector	/XCODES/(	180)	ADEQ3A	I	LB
						ADICB3	O	LB
						ADIC3A	M	NN
MB		O	Last element number in partitioned $d\Psi$ vector	/XCODES/(	181)	ADEQ3A	I	MB
						ADICB3	O	MB
						ADIC3A	O	MB
NB		M	Extent of integration set during adjoints on branch problem	/XCODES/(	179)	ADEQ3A	I	NB
						ADICB3	M	NB
						ADIC3A	M	NB
						BNTG	O	NB
						RKTB3A	I	NB
						STVRL3	I	NB
NCN		I	Number of elements in $d\Psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BSTO3	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAYO2	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NCTIN		I	Number of elements in upper triangular portion of A matrix	/XCODES/(	184)	ADICB3	I	NCTIN
						ADIC3A	M	NCTIN
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BSTO3	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
NEQB		I	Number of integrated quantities during adjoint solution	/XCODES/(	161)	ADICB3	I	NEQB
						ADIC3A	M	NEQB
						RKTB3A	I	NN
NICNB		I	Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
						ADIC3A	I	NICNB
						BNTG	I	NICNB
						REU3	I	NICNB
						SDINP	M	NICNB
						TEST	I	NICNB
						TRAN3	I	NICNB
NSAB		I	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB
SFD	$\dot{\Omega}$	I	Rate of change of cut-off function	/ADICB3/(*	)	ADICB3	I	SFD
SVBV		I	Saved state vector on trial trajectory	/STATE3/(	710)	ADICB3	I	SVBV
						REU3	O	SVBV
SX	$\partial\Omega/\partial y$	I	Partial of cut-off WRT state	/ADICB3/(*	)	ADICB3	I	SX
TIME	t	O	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
TST		I	Array of arc end times on trial trajectory [sd]	/GENF /(	331)	ADICB3	I	TST
						BNTG	I	TST
						FNTG	O	TST
						TEST	I	TST
VAR	v	O	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	D	KWOW
						YREF3	M	V
VARL		O	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A	I	VARL
						ADICB3	O	VARL
						ADIC3A	O	VARL
						PROPIN	I	ZZ
						RKTA3A	M	F
						RKTB3A	M	Y
						STVRL3	O	VARL
						TRAN3	M	VARL

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
XL	$\lambda^{\Psi_i \Omega_j}$	M	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
YDS	$\dot{y}_{i,\gamma-}$	I	Array of state derivatives at arc end points	/STATE3/(	507)	ADICB3	I	YDS
						ADID3A	I	YDS
						REV3	O	YDS
						STAU	I	YDS

```
      SUBROUTINE ADICB3                                                 ADICB3
C        THIS ROUTINE HAS THE FOLLOWING  ENTRY/PURPOSE                  COMN
C        ENTRY ADI3B/1 SAVE  A AT INITIAL POINT OF BRANCH 3             COMN
C                    2 RESET STATE AND STATE DER. TO VALUE AT END       COMN
C                      OF BRANCH 2                                      COMN
C                    3 SET RANGE OF INTEGRTED ADJOINTS                  COMN
C                    4 COMP  ADJ INITIAL COND AT END OF BRANCH 2        COMN
C        ENTRY ADICB/1 SETS UP NECESSARY DATA FOR INITIALIZING          COMN
C                      ADJOINTS AT END OF INTERMEDIATE CONSTRAINT ARC   COMN
C                    2  COMPUTES ADJOINT INITIAL COND.                  COMN
C        ENTRY ADI2B/1  RESTORES A  TO FULL SIZE AT END OF TRUNK ARC    COMN
C                    2  RESETS RANGE OF ADJOINT INTEGRATION             COMN
C                                                                       COMN
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI    ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    , STATE3D
     *SVBY   (9),OMEGA     ,OMEGA2    ,                                 STATE3D
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP       ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO       ,UDO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                      STATE3D
      COMMON/GENF/                                                      GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP       , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED    ,IRATED    ,                                            GENF
     *P1        ,P2        ,P3        ,XK1       ,XK2       ,XK3       , GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
     *I2OP      ,ICOP      ,IFA       ,IFAR      ,IFB       ,IND       , XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
     *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
     *NCN       ,NEQB      ,NEG       ,NOP       ,NPH       ,N         , XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN     ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      DIMENSION SX(7)                                                   ADICB3
C                                                                       COMN
C        I   BRANCH POINT APPROACHED FROM BRANCH 3                      COMN
      ENTRY  ADI3B                                                      ADICB3
```

```
      ISKP = 0                                                          ADICB3
C       I-A  SAVE A                                                     COMN
      DO 10 I=1,NCN                                                     ADICB3
      DO 10 J=1,NCN                                                     ADICB3
   10 COTI(J,I)=A(J,I)                                                  ADICB3
C       I-B  RESET TIME AND STATE TO END OF BRANCH 2                    COMN
      NS =  NSB + NSAB                                                  ADICB3
      NB =  (NCN -NICNB)*4 +1                                           ADICB3
      TIME=  TST(NS)                                                    ADICB3
      DO 20  I=1,NEQ                                                    ADICB3
      VAR(I) =SVBV(I)                                                   ADICB3
   20 DVAR(I)=YDS(NS,I)                                                 ADICB3
C       I-C  SET RANGE OF CONSTRAINT VECTOR AND ADJOINT INTEGRATION     COMN
C        ARRAY                                                          COMN
      MB=NICNB                                                          ADICB3
      LB=1                                                              ADICB3
      NX =  NEQB -NCTIN                                                 ADICB3
C      II     DETERMINE  CUTOFF VARIABLE CODE,IF STATE,II-A,            COMN
C             ELSE CALC.PARTIALS                                        COMN
   30 JS =IABS(JST(NS))                                                 ADICB3
      IF(JS.LT.9) GO TO 40                                              ADICB3
      CALL PDBC(JS,DVAR,SX,SFD,3,ISKP)                                  ADICB3
   40 CONTINUE                                                          ADICB3
C      II-A   ZERO ADJOINT INTEGRATION ARRAY                            COMN
      DO 50  I=NB,NX                                                    ADICB3
      VARL(I)=0.                                                        ADICB3
   50 DVARL(I)=0.                                                       ADICB3
   60 CONTINUE                                                          ADICB3
C      III     COMPUTE ADJOINT  INITIAL CONDITIONS                      COMN
      DO 140   I=1,NICNB                                                ADICB3
      DO 70 KM=1,NEQ                                                    ADICB3
   70 XL(KM,I)=0.                                                       ADICB3
C      III-A    DETER. CONST. CODE  IF NOT TIME ,III-B                  COMN
C               ELSE  SET  DFD=1.,III-D                                 COMN
      II =  ITQ(I)                                                      ADICB3
      IF(II.GT.1) GO TO 80                                              ADICB3
      DFD= 1.                                                           ADICB3
      GO TO  110                                                        ADICB3
C      III-B    IF CONST IS FUNCTION,III-C  ,ELSE SET PROPER ADJOINT    COMN
C               EQUAL TO 1. AND  CALC.DFD                               COMN
   80 IF(II.GT.10) GO TO 100                                            ADICB3
      IF(II.EQ.10) GO TO 90                                             ADICB3
      XL(II-1,I) =1.                                                    ADICB3
      DFD =  YDS(NS,II-1)                                               ADICB3
      GO TO 110                                                         ADICB3
   90 XL(8,I)=1.                                                        ADICB3
      DFD= YDS(NS,8)                                                    ADICB3
      GO TO 110                                                         ADICB3
  100 CALL PDBC(II,DVAR,XL(1,I),DFD,3,ISKP)                             ADICB3
C      III-C    CONST.IS FUNCT.,COMPUTE PARTIALS AND DFD                COMN
  110 CONTINUE                                                          ADICB3
C      III-D      COMBINE CONST. AND CUT-OFF TERMS INTO ADJOINT         COMN
C                 INITIAL CONDITIONS                                    COMN
      IF(JS.LE.2) GO TO 140                                             ADICB3
      IF(JS.GT.9) GO TO 120                                             ADICB3
      XL(JS-2,I)=    XL(JS-2,I) -DFD /YDS(NS,JS-2)                      ADICB3
      GO TO 140                                                         ADICB3
  120 DO 130 KK=1,7                                                     ADICB3
  130 XL(KK,I) = XL(KK,I) - DFD/SFD *SX(KK)                             ADICB3
  140 CONTINUE                                                          ADICB3
      RETURN                                                            ADICB3
      ENTRY  ADICB                                                      ADICB3
C       IV      AT  INTEMEDIATE ARC CONSTRAINT, SET  STATE DERIV        COMN
C               EQUAL TO STORED NOMINAL VALUE                           COMN
      NS= IARC-1                                                        ADICB3
      DO 150 I=1,NEQ                                                    ADICB3
  150 DVAR(I)= YDS(NS,I)                                                ADICB3
      NB =1                                                             ADICB3
      LB=1                                                              ADICB3
      MB=NCN                                                            ADICB3
      NX =  4* NICNB                                                    ADICB3
```

20 OCT 72 G 01-46

```
      GO TO  30                                                        ADICB3   30
      ENTRY ADI2B                                                      ADICB3
C       V       RESTORE  A MATRIX  AND ITS INTEGRAL FOR                COMN
C                 CONTINUING BACKWARD ADJOINT SOLUTION ON TRUNK         COMN
      MB=NCN                                                           ADICB3
      NX=NICNB+1                                                       ADICB3
      DO 160 I=NX,NCN                                                  ADICB3
      DO 160 J= 1,NCN                                                  ADICB3
      A(I,J)=COTI(I,J)                                                 ADICB3
  160 A(J,I)=COTI(J,I)                                                 ADICB3
      NI=NEQB-NCTIN                                                    ADICB3
      DO 170 I=1,NCN                                                   ADICB3
      DO 170 J=1,I                                                     ADICB3
      NI=NI+1                                                          ADICB3
  170 VARL(NI)=A(J,I)                                                  ADICB3
      RETURN                                                           ADICB3
      END                                                              ADICB3
```

20 OCT 72 6.01-46

SUBRØUTINE
ADIC3A

Subroutine ADIC3A

Purpose

Subroutine ADIC3A (entry point ADIC) computes adjoint initial conditions at the terminus of the trajectory. After zeroing out the adjoint variable matrix, the impulse response functions and the adjoint integration array (VARL and PVARL), equation (11.1-23) of Volume I is solved.

Description

This routine is called from TEST.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DFD	$\dot{\Psi}$	W	Rate of change of constraint	/ADIC3A/(*	)	ADIC3A	W	DFD
DVAR	$\ddot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
DVARL		O	Array of derivatives for adjoint integration	/STATE3/(	128)	ADEQ3A	O	DVARL
						ADICB3	O	DVARL
						ADIC3A	O	DVARL
						RKTB3A	I	DY
INTB		I	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
						BNTG	I	INTB
						ENVPRM	I	INTB
						FNTG	I	INTB
						SDINP	M	INTB
						TEST	I	INTB
						TRAN3	I	INTB
						TRTOSZ	I	INTB
ISKP		W	Flag to stop redundant computation in subroutine PDBC	/ADIC3A/(*	)	ADIC3A	W	ISKP
ITQ		I	Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
						ADIC3A	I	ITQ
						ADID3A	I	ITQ
						CON3	I	ITQ
						SDINP	M	ITQ
						STAU	I	ITQ
						TOPM	O	IITQ
JK		M	Integration routine flag tells which derivative evaluation in Runge-Kutta cycle	/XCODES/(	151)	ADIC3A	M	JK
						BNTG	I	JK
						MODELA	I	JK
						PAYO2	M	JK
						RKTA3A	M	J
						RKTB3A	M	J
JS		I	Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
						ADIC3A	I	JS
						ADID3A	I	JS
						BNTG	M	JS
						FNTG	M	JS
						PROPB	I	JS
						PROPIN	I	JS
						STP3	I	JS
						TOL3	I	JS
MB		O	Last element number in partitioned dΨ vector	/XCODES/(	181)	ADEQ3A	I	MB
						ADICB3	O	MB
						ADIC3A	O	MB
NB		M	Extent of integration set during adjoints on branch problem	/XCODES/(	179)	ADEQ3A	I	NB
						ADICB3	M	NB
						ADIC3A	M	NB
						BNTG	O	NB
						RKTB3A	I	NB
						STVRL3	I	NB

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NCN		I	Number of elements in $d\Psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BST03	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAY02	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NCTIN		M	Number of elements in upper triangular portion of A matrix	/XCODES/(	184)	ADICB3	I	NCTIN
						ADIC3A	M	NCTIN
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BST03	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
NEQB		M	Number of integrated quantities during adjoint solution	/XCODES/(	161)	ADICB3	I	NEQB
						ADIC3A	M	NEQB
						RKTB3A	I	NN
NICNB		I	Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
						ADIC3A	I	NICNB
						BNTG	I	NICNB
						REU3	I	NICNB
						SDINP	M	NICNB
						TEST	I	NICNB
						TRAN3	I	NICNB
NN		M	First element number in partitioned $d\Psi$ vector	/XCODES/(	180)	ADEQ3A	I	LB
						ADICB3	O	LB
						ADIC3A	M	NN
SFD	$\dot{\Omega}$	I	Rate of change of cut-off function	/ADIC3A/(*	)	ADIC3A	I	SFD
SX	$\partial\Omega/\partial y$	I	Partial of cut-off WRT state	/ADIC3A/(*	)	ADIC3A	I	SX
VARL		O	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A	I	VARL
						ADICB3	O	VARL
						ADIC3A	O	VARL
						PROPIN	I	ZZ
						RKTA3A	M	F
						RKTB3A	M	Y
						STVRL3	O	VARL
						TRAN3	M	VARL
XL	λ^{Ψ}, Ω_J	M	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BST03	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	*	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
XLAMA	$\Lambda^{\Psi_i \Omega_j}$	0	Impulse response function column vector associated with angle of attack		/AECO3 /(	16)	ADEQ3A	M	XLAMA
							ADIC3A	O	XLAMA
							AST3	O	XLAMA
							BGET3	O	XLAMA
							BSTO3	M	XLAMA
							MTX3A	I	XLAMA
							TRAN3	M	XLAMA
XLAMP	$\Lambda^{\Psi_i \Omega_j}$	0	Impulse response function column vector associated with bank angle		/AECO3 /(	25)	ADEQ3A	M	XLAMP
							ADIC3A	O	XLAMP
							AST3	O	XLAMP
							BGET3	O	XLAMP
							BSTO3	M	XLAMP
							MTX3A	I	XLAMP
							TRAN3	M	XLAMP

30 OCT 72 G 01-46

```
      SUBROUTINE ADIC3A                                                    ADIC3A
C          COMPUTES   ADJOINT  INITIAL CONDITIONS AT                       COMN
C          TRAJECTORY  TERMINUS                                            COMN
      DIMENSION  SX(7)                                                     ADIC3A
      COMMON/STATE3/                                                       STATE3D
     *VAR(14)     ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)      ,SVY(10)   , STATE3D
     *XL(9,9)     ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15) , STATE3D
     *SINPSI      ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2   , STATE3D
     *SVBV    (9),OMEGA       ,OMEGA2     ,                                 STATE3D
     *VDV         ,GDV        ,RDV        ,MDV        ,PDV        ,ODV      , STATE3D
     *UDV         ,VDG        ,GDG        ,RDG        ,PDG        ,ODG      , STATE3D
     *UDG         ,VDR        ,GDR        ,MDR        ,PDR        ,ODR      , STATE3D
     *UDR         ,VDM        ,GDM        ,MDM        ,PDM        ,VDP      , STATE3D
     *GDP         ,PDP        ,ODP        ,UDP        ,VDO        ,GOO      , STATE3D
     *PDO         ,UDO        ,HTDV       ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                                  STATE3D
      COMMON/STATE3/                                                       STATE3D
     *SIN2RO      ,COS2RO     ,COS2GM                                      STATE3D
      COMMON/AECO3/                                                        AECO3
     *APHO        ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA      , AECO3
     *SINA        ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI   , AECO3
     *COSPHI      ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDO      , AECO3
     *CDOM        ,CLO        ,FK         ,XCGM       ,ZCGM       ,CLOM     , AECO3
     *CM          ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM       ,FKM      , AECO3
     *CLAM        ,CL         ,CLA        ,CLM        ,                       AECO3
     *CD          ,CDA        ,CDM                                          AECO3
      EQUIVALENCE(LB,NN)                                                   ADIC3A
      COMMON /XCODES/                                                      XCODES
     *ITQ     (9),ICOR  (20),ITI        ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)              ,NCNST      ,NSB        ,NSAB       ,NICNB    , XCODES
     *I20P        ,ICOP       ,IFAW       ,IFAR       ,IFB        ,IND      , XCODES
     *IOPEN       ,IPH        ,ISPH       ,ISST       ,IARC       ,ISTART   , XCODES
     *ITCT        ,ITER       ,IVAR       ,JK         ,JPS        ,JS       , XCODES
     *KOP         ,KPST       ,K          ,KST        ,NAD        ,NCASE    , XLODES
     *NCN         ,NEQB       ,NEQ        ,NOP        ,NPH        ,N        , XCODES
     *NST         ,IPST       ,IPRINT     ,ISTN       ,IPHN       ,ISTNB    , XCODES
     *IPHNB       ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP      ,L        , XCODES
     *IFOB        ,NB         ,LB         ,MB         ,NPHP       ,NPHB     , XCODES
     *NCTIN       ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3            XCODES
      COMMON/GENF/                                                         GENF
     *OMG(20)     ,OMGP(20,2),VARQ(9)    ,TOL(9)     ,SVAR(10)   ,WDC(20)  , GENF
     *A(9,9)      ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)    ,DTP      , GENF
     *DTS         ,DT         ,G          ,DPSQ       ,Q          ,QS       , GENF
     *R           ,RE         ,MACH       ,PA         ,RO         ,CS       , GENF
     *VNU         ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ    , GENF
     *SVSQ        ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)    , GENF
     *TST(20)     ,TPH    (20),DIS(20)    ,DIP(20)    ,T          ,W        , GENF
     *TLP1(20)    ,TLS1   (20),DIP1(20)   ,DIS1(20)   ,TIME       ,OMP      , GENF
     *TIMPR       ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)  , GENF
     *AE          ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV    , GENF
     *QR          ,QV         ,FVAC       ,LIFTV      ,                       GENF
     *LIFTR       ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA    , GENF
     *             LIFTM      ,DBR        ,DB         ,ISP        ,ISPF     , GENF
     *                         ULFT       ,ULFTV      ,ULFTR      ,ULFTA    , GENF
     *XMCG        ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE    , GENF
     *CULFT       ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SID      , GENF
     *COD         ,SIDAE      ,XCG        ,ZCG        ,XJ                     GENF
      COMMON / GENF  /                                                     GENF
     *XJV         ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP      , GENF
     *FRATED    ,IRATED       ,                                             GENF
     *P1          ,P2         ,P3       ,XK1          ,XK2        ,XK3      , GENF
     *XK1T        ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D     , GENF
     *XK1A        ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V     , GENF
     *XK1G        ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P     , GENF
     *XK1R        ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O     , GENF
     *XK1U        ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M     , GENF
     *PV          ,PG         ,PP         ,PR         ,PO         ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,             MACH, MACHR,           GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                      FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      ENTRY ADIC                                                           ADIC3A
      ISKP=0                                                               ADIC3A
C        I            **            **          **                         COMN
C  ZERO XL ,DVARL,VARL,XLAMA  ARRAYS  AFTER CALCULATING INDICES            ADIC3A
```

20 OCT 72 6.01-46

```
      NCTIN = (NCN*(NCN+1))/2                                          ADIC3A
      NEQB= 6*NCN + NCTIN                                              ADIC3A
C       I-A     CHECK  FOR  BRANCHING  OR  INTEMEDIATE CONSTRAINTS     COMN
C             TO  COMPUTE  RANGE OF INTEGRATION  ARRAY                 COMN
      IF(INTB-1)10,20,20                                               ADIC3A
   10 NB =1                                                            ADIC3A
      NN= 1                                                            ADIC3A
      GO TO  30                                                        ADIC3A
   20 NB = 4*NICNB+1                                                   ADIC3A
      NN = NICNB +1                                                    ADIC3A
   30 DO 40 I=1,NCN                                                    ADIC3A
      XLAMP(I)=0.                                                      ADIC3A
      XLAMA(I)=0.                                                      ADIC3A
      DO 40 J=1,NEQ                                                    ADIC3A
      XL(J,I)=0.                                                       ADIC3A
   40 CONTINUE                                                         ADIC3A
      DO 50 I=1,NEQB                                                   ADIC3A
      VARL(I)=0                                                        ADIC3A
      DVARL(I)=0.                                                      ADIC3A
   50 CONTINUE                                                         ADIC3A
C       II      COMPUTE ADJOINT  INITIAL CONDITIONS                    COMN
C                                                                      COMN
C       II-A   TEST  FOR   LINEAR  CUT-OFF ,YES,SKIP NON-LINEAR        COMN
C               CUTOFF CALCULATIONS                                    COMN
      IF(JS.LE.9) GO TO 60                                             ADIC3A
      CALL PDBC(JS,DVAR,SX,SFD,3,ISKP)                                 ADIC3A
   60 CONTINUE                                                         ADIC3A
      DO 130 I=NN,NCN                                                  ADIC3A
      II=ITQ(I)                                                        ADIC3A
C       II-B     SET-UP  OR CALCULATE  RATE OF CHANGE  OF CONSTRAINT   COMN
C       AND  IC  FOR  LINEAR  CONSTRAINTS                              COMN
      IF(II.GT.1) GO TO 70                                             ADIC3A
      DFD=1.                                                           ADIC3A
      GO TO  100                                                       ADIC3A
   70 IF(II.GT.10) GO TO90                                             ADIC3A
      IF(II.EQ.10) GO TO 80                                            ADIC3A
      XL(II-1,I)= 1.                                                   ADIC3A
      DFD = DVAR(II-1)                                                 ADIC3A
      GO TO 100                                                        ADIC3A
   80 XL(8,I)=1.                                                       ADIC3A
      DFD=DVAR(8)                                                      ADIC3A
      GO TO 100                                                        ADIC3A
C       II-C     SET-UP  AND  CALCULATE  IC  FOR NON-LINEAR CONSTRAINT COMN
   90 CALL  PDBC(II,DVAR,XL(1,I),DFD,3,ISKP)                           ADIC3A
  100 CONTINUE                                                         ADIC3A
      IF(JS.LE.2) GO TO 130                                            ADIC3A
      IF(JS.GT.9) GO TO 110                                            ADIC3A
      XL(JS-2,I) = XL(JS-2,I) -DFD/DVAR(JS-2)                          ADIC3A
      GO  TO  130                                                      ADIC3A
  110 DO 120 KK=1,7                                                    ADIC3A
  120 XL(KK,I) = XL(KK,I) - DFD/SFD*SX(KK)                             ADIC3A
  130 CONTINUE                                                         ADIC3A
      JK=NB                                                            ADIC3A
C       III    PRINT    ADJOINT  ICS    AND RETURN                     COMN
      DO 150 I=NN,NCN                                                  ADIC3A
      DO 140 JJ=1,6                                                    ADIC3A
      VARL(JK) = XL(JJ,I)                                              ADIC3A
  140 JK=JK+1                                                          ADIC3A
  150 CONTINUE                                                         ADIC3A
      CALL IPR(10HADJ,IN,CND,XL(1,NN),I,( NCN-NN+1)*9, 0)              ADIC3A
      NB=NCN                                                           ADIC3A
      RETURN                                                           ADIC3A
      END                                                              ADIC3A
```

20 OCT 72 G 01-46

SUBRØUTINE ADID3A

Subroutine ADID3A

Purpose

Subroutine ADID3A computes adjoint discontinuities for different situations using different entry points. It accounts for branching and intermediate constraints as well as most mixed boundary conditions. Mixed boundary condition adjoint discontinuities are handled for elapsed time and functions of elapsed time only.

Description

Preliminary calculations at each entry point use the local range of the $d\psi$ vector to determine which adjoint sets are to be considered.

It should be noted that this subroutine is not called if the cut-off function is on fixed arc time. This routine's entry points are called from BNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DVAR	$\dot{y}$	M	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTDSZ	I	IARC
IPH		I	Phase number	/XCODES/(	143)	ADID3A	I	IPH
						ADJUST	I	IPH
						AST3	I	IPH
						BNTG	M	IPH
						FNTG	M	IPH
						GETIT	I	IPH
						GUI3A	I	IPH
						SDINP	M	IPH
ISKP		W	Flag to stop redundant computation in subroutine PDBC	/ADID3A/(*	)	ADID3A	W	ISKP
ITQ		I	Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
						ADIC3A	I	ITQ
						ADID3A	I	ITQ
						CON3	I	ITQ
						SDINP	M	ITQ
						STAU	I	ITQ
						TOPM	D	IITQ
JPS		I	Absolute value of phase cut-off option code	/XCODES/(	152)	ADID3A	I	JPS
						BNTG	M	JPS
						FNTG	M	JPS
						STP3	I	JPS
						TOL3	I	JPS
JS		I	Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
						ADIC3A	I	JS
						ADID3A	I	JS
						BNTG	M	JS
						FNTG	M	JS
						PROPB	I	JS
						PROPIN	I	JS
						STP3	I	JS
						TOL3	I	JS

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NCN		I	Number of elements in $d\Psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BST03	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAY02	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BST03	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
OMGZ	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
						CRASH	I	OMEGA
						DER3A	I	OMGZ
						EQUA3	I	OMGZ
						GEINP	I	OMGZ
						MODELA	I	OMGZ
						MODELB	I	OMGZ
						PDBC	I	OMGZ
						PDY3A	I	OMGZ
						SDINP	I	OMGZ
						TOPM	I	OMGZ
SFD	$\dot{\Omega}$	I	Rate of change of cut-off function	/ADID3A/(*	)	ADID3A	I	SFD
SX	$\partial\Omega/\partial y$	I	Partial of cut-off WRT state	/ADID3A/(*	)	ADID3A	I	SX
XL	λ^{Ψ,Ω_J}	M	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BST03	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
YDP		I	Array of state derivatives at phase end points	/STATE3/(	327)	ADID3A	I	YDP
						REU3	O	YDP
YDS	$\dot{y}\vert_{\tau-}$	I	Array of state derivatives at arc end points	/STATE3/(	507)	ADICB3	I	YDS
						ADID3A	I	YDS
						REU3	O	YDS
						STAU	I	YDS

```
      SUBROUTINE ADID3A(JA)                                                 ADID3A
C         COMPUTES  ADJOINT  DISCONTINUITIES  FOR  DIFFERENT                CORN
C         SITUATIONS  USING  DIFFERENT  ENTRY  POINTS                       COMN
C         ENTRY  ADIDS= BRANCH 2  APPROACHING TRUNK  OR INTER  CONST  AR    CORN
C         ENTRY  ADIDSS= NORMAL MODE                                        CORN
C         ENTRY  ADIDP = PHASE  CORNER NORMAL MODE                          CORN
C                                                                           CORN
      COMMON/GLOBAL/                                                        GLOBAL
     *GR          ,ER          ,OMGZ      ,XLAMRF    ,YMUPF     ,LUM        GLOBAL
     *,JJOP(10) ,IFATAL    ,NARC       ,NBRAN     ,NFARC     ,ID(4)         GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                                   GLOBAL
     *,GM         ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)        GLOBAL
     *,ITPS0  ,KSOL     ,KGLOBL(8)                                          RETAP
      COMMON/GENF/                                                          GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   ,   GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       ,  GENF
     *DTS        ,DT         ,G         ,DPSQ      ,Q         ,QS        ,  GENF
     *R          ,RE         ,MACH      ,PA        ,RO        ,CS        ,  GENF
     *VNU        ,PAR        ,ROR       ,CSR       ,VNR       ,SUMSQ     ,  GENF
     *SVSQ       ,TIMEPH     ,TIMES     ,TOP       ,TOS       ,TR(9)     ,  GENF
     *TST(20)    ,TPH    (20),DIS(20)   ,DIP(20)   ,T         ,W         ,  GENF
     *TLP1(20)   ,TLS1   (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       ,  GENF
     *TIMPR      ,LIFT       ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,  GENF
     *AE         ,FP         ,FPOLD     ,FPD       ,MACHR     ,MACHV     ,  GENF
     *QR         ,QV    '    ,FVAC      ,LIFTV     ,                        GENF
     *LIFTR      ,LIFTA      ,           DRAGV     ,DRAGR     ,DRAGA     ,  GENF
     *            LIFTM      ,DBR       ,DB        ,ISP       ,ISPF      ,  GENF
     *                        ULFT      ,ULFTV     ,ULFTR     ,ULFTA     ,  GENF
     *XMCG       ,XMCGV      ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,  GENF
     *CULFT      ,CT         ,CALPHA    ,CDE       ,DELTAE    ,SID       ,  GENF
     *COD        ,SIDAE      ,XCG       ,ZCG       ,XJ                      GENF
      COMMON / GENF  /                                                      GENF
     *XJV         XJR        ,GH        ,GAMMAD    ,XKG       ,XKP       ,  GENF
     *FRATED   ,IRATED      ,                                               GENF
     *P1         ,P2         ,P3       ,XK1        ,XK2       ,XK3       ,  GENF
     *XK1T       ,XK2T       ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,  GENF
     *XK1A       ,XK2A       ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,  GENF
     *XK1G       ,XK2G       ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,  GENF
     *XK1R       ,XK2R       ,XK3R      ,XK1O      ,XK2O      ,XK3O      ,  GENF
     *XK1U       ,XK2U       ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,  GENF
     *PV         ,PG         ,PP        ,PR        ,PO        ,DPDY(3,8)    GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                      FRAT
      DIMENSION TPH1(10),TST1(10)                                           GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                    GENF
      COMMON /XCODES/                                                       XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2),   XCODES
     *JST    (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB     ,  XCODES
     *I2OP       ,ICOP       ,IFAW      ,IFAR      ,IFB       ,IND       ,  XCODES
     *IOPEN      ,IPH        ,ISPH      ,ISST      ,IARC      ,ISTART    ,  XCODES
     *ITCT       ,ITER       ,IVAR      ,JK        ,JPS       ,JS        ,  XCODES
     *KOP        ,KPST       ,K         ,KST       ,NAD       ,NCASE     ,  XCODES
     *NCN        ,NEQB       ,NEQ       ,NOP       ,NPH       ,N         ,  XCODES
     *NST        ,IPST       ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     ,  XCODES
     *IPHNB      ,IBLK1      ,IBLK2     ,ISTOP     ,ISTPP     ,L         ,  XCODES
     *IFOB       ,NB         ,LB        ,MB        ,NPHP      ,NPHB      ,  XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON/STATE3/                                                        STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)      ,SVY(10)   ,  STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15) ,  STATE3D
     *SINPSI     ,COSPSI     ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    ,  STATE3D
     *SVBV    (9),OMEGA      ,OMEGA2    ,                                   STATE3D
     *VDV        ,GDV        ,RDV       ,MDV       ,PDV       ,ODV       ,  STATE3D
     *UDV        ,VDG        ,GDG       ,RDG       ,PDG       ,ODG       ,  STATE3D
     *UDG        ,VDR        ,GDR       ,MDR       ,PDR       ,ODR       ,  STATE3D
     *UDR        ,VDM        ,GDM       ,MDM       ,PDM       ,VDP       ,  STATE3D
     *GDP        ,PDP        ,ODP       ,UDP       ,VDO       ,GDO       ,  STATE3D
     *PDO        ,UDO        ,HTDV      ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                   STATE3D
      COMMON/STATE3/                                                        STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                        STATE3D
      DIMENSION  SX(7)                                                      ADID3A
C       I     SET UP  FOR  SECOND BRANCH                                    CORN
      ENTRY  ADIOS                                                          ADID3A
      ISKP=0                                                                ADID3A
```

20 OCT 72 G 01-46

```
      N1= JA                                                      ADID3A
      N2= NCN                                                     ADID3A
      GO TO  10                                                   ADID3A
      ENTRY ADIDSS                                                ADID3A
C       II   SET  UP FOR   NORMAL  MODE                           COMN
      ISKP=0                                                      ADID3A
      N1=1                                                        ADID3A
      N2=NCN                                                      ADID3A
   10 CONTINUE                                                    ADID3A
      KIK=1                                                       ADID3A
      IF(JS.LT.9) GO TO  40                                       ADID3A
      KK=JS                                                       ADID3A
C       III    COMPUTE  PARTIALS  FOR  CUTOFF FUNCTION            COMN
      DO 20 I=1,NEQ                                               ADID3A
   20 DVAR(I)= YDS(IARC-1,I)                                      ADID3A
   30 CALL  PDBC(KK,DVAR,LX,SFD,3,ISKP )                          ADID3A
   40 GO TO (50,100),KIK                                          ADID3A
   50 CONTINUE                                                    ADID3A
C       IV   COMPUTE  ADJOINT DISCONTINUITY  FOR ARC CORNER       COMN
C                                                                 COMN
      DO 90 I=N1,N2                                               ADID3A
      II = ITQ( I)                                                ADID3A
      TT = 0.                                                     ADID3A
      IF(II.EQ.1) TT=1.                                           ADID3A
      IF(II EQ 14.OR II EQ 20 ) TT =OMGZ                          ADID3A
      DO 60  IK=1,NEQ                                             ADID3A
   60 TT = TT + XL(IK,I)* YDS(IARC-1,IK)                          ADID3A
      IF(JS.GT. 9) GO TO 70                                       ADID3A
      XL(JS-2,I) = XL(JS-2,I) - TT / YDS(IARC-1,JS-2)             ADID3A
      GO TO 90                                                    ADID3A
   70 DO 80 JJ=1,7                                                ADID3A
   80 XL(JJ,I) = XL(JJ,I) - TT/SFD*SX(JJ)                         ADID3A
   90 CONTINUE                                                    ADID3A
      RETURN                                                      ADID3A
C                                                                 COMN
C       V    COMPUTE  ADJOINT  DISCONTINUITY  FOR  PHASE  CORNER  COMN
      ENTRY  ADIDP                                                ADID3A
      IF(JPS.LT.9) GO TO 100                                      ADID3A
      KK= JPS                                                     ADID3A
      KIK=2                                                       ADID3A
      GO TO  30                                                   ADID3A
  100 CONTINUE                                                    ADID3A
      DO 140  I=JA,NCN                                            ADID3A
      TT =0.                                                      ADID3A
      DO 110  IK=1,NEQ                                            ADID3A
  110 TT = TT + XL(IK,I) *YDP(IPH-1,IK)                           ADID3A
      IF(JPS.GT.9) GO TO 120                                      ADID3A
      XL(JPS-2,I) = XL(JPS-2,I) -TT / YDP(IPH-1,JPS-2)            ADID3A
      GO TO 140                                                   ADID3A
  120 DO 130 JJ=1,7                                               ADID3A
  130 XL(JJ,I) = XL(JJ,I) - TT /SFD *SX(JJ)                       ADID3A
  140 CONTINUE                                                    ADID3A
      RETURN                                                      ADID3A
      END                                                         ADID3A
```

SUBRØUTINE ADJUST

Subroutine ADJUST

Purpose

Subroutine ADJUST has several entry points; each is described below.

Description

Entry ADJUS. This entry adjusts initial states that are to be optimized. It will also set the arc time optimization flag, ITI, if the first arc time duration is to be optimized. After computing the adjustable parameter increment in MTX1, the free initial state is adjusted and the sensitivity is removed from the sensitivity matrix. The number of free parameters is also reduced appropriately.

Entry ADJIN. This entry checks for arc time duration optimization in the prior arc. If it occurs, the logic eliminates the appropriate elements in the sensitivity matrix and reduces the number of free parameters by one.

The entry points in this routine are called by FNTG.

Entry ADJUT. This entry adjusts the free arc time cut-off value using the currently computed value of the parameter correction. It also forces a concurrent phase-time cut-off to occur at the same time as the arc-time cut-off.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DPAR	δp	I	Adjustable parameter corrections	/PARAM /(	264)	ADJUST	I	DPAR
						MTX3A	M	DPAR
						TOPM	O	DPAR
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
ICOR		I	Phase sequence array	/XCODES/(	10)	ADJUST	I	ICOR
						FNTG	I	ICOR
						PRMSET	I	ICOR
						SDINP	M	ICOR
IPH		I	Phase number	/XCODES/(	143)	ADID3A	I	IPH
						ADJUST	I	IPH
						AST3	I	IPH
						BNTG	M	IPH
						FNTG	M	IPH
						GETIT	I	IPH
						GUI3A	I	IPH
						SDINP	M	IPH
IPOINT		I	Code for each adjustable parameter in steepest descent.	/PARAM /(	1)	ADJUST	I	IPOINT
						PRMSET	I	IPOINT
						SDINP	O	IPOINT
						STAU	I	IPOINT
						TOPM	O	IPOINT
ITI		M	Optimized arc time flag	/XCODES/(	30)	ADJUST	M	ITI
						FNTG	I	ITI
						SDINP	O	ITI
KST		I	Arc or phase cut-off flag	/XCODES/(	157)	ADJUST	I	KST
						FNTG	M	KST
NCN		I	Number of elements in $d\Psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BSTO3	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAYO2	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NPAR		M	Running count of number of adjustable parameters to be perturbed on remainder of trajectory	/PARAM /(	14)	ADJUST	M	NPAR
						FNTG	I	NPA
						MTX3A	I	NPA
						TOPM	O	NPA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NPARA		I	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAY02	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	D	NPARA
OMG	Ω_J	M	Array of arc cut off values [sd]	/GENF /(	1)	ADJUST	M	OMG
						FNTG	I	OMG
						PRMSET	M	OMG
						PROPB	I	OMG
						SDINP	M	OMG
						STP3	I	OMG
						TOPM	D	IOMG
OMGP		O	Array of phase cut off values [sd]	/GENF /(	21)	ADJUST	O	OMGP
						FNTG	M	OMGP
						PRMSET	O	OMGP
						SDINP	M	OMGP
OMP		O	Phase cut-off value	/GENF /(	494)	ADJUST	O	OMP
						FNTG	M	OMP
						STP3	I	OMP
PARA	p	I	Adjustable parameter nominal values.	/PARAM /(	252)	ADJUST	I	PARA
						PRMSET	M	PARA
						TOPM	D	PARA
SPARA	S^{Ψ_i}	I	Matrix of adjustable parameter sensitivities (including all parameters)	/PARAM /(	15)	ADJUST	I	SPARA
						PAY02	I	SPARA
						STAU	M	SPARA
						TOPM	D	SPARA
SPARB	S^{Ψ_I}	O	Matrix of adjustable parameter sensitivities (Contains only elements corresponding to parameters yet to be adjusted)	/PARAM /(	144)	ADJUST	O	SPARB
						MTX3A	I	SPARB
						TOPM	D	SPARB
SVAR	$y\|_{t=0}$	O	Array of state values at initial problem time [sd]	/GENF /(	79)	ADJUST	O	SVAR
						BNTG	I	SVAR
						FNTG	I	SVAR
						PRMSET	M	SVAR
						REU3	I	SVAR
						SDINP	M	SVAR
						TEST	I	SVAR
						TOPM	I	SVAR
						TRTOSZ	I	SVAR
S2INV	[SS]	O	Parameter sensitivity contribution to A matrix	/PARAM /(	276)	ADJUST	O	S2INV
						MTX3A	M	S2INV
						PAY02	M	S2INV
						TOPM	D	S2INV

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
VAR	v	M	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	D	VAR
							ADJUST	M	VAR
							AGETB3	D	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	V
							STP3	I	VAR
							TOPM	D	KWDW
							YREF3	M	V

```
      SUBROUTINE ADJUST                                                  ADJUST
C                                                                        COMN
C           COMPUTES   ADJUSTMENTS  IN  OPTIMIZED INITIAL  STATES        COMN
      COMMON/GENF/                                                       GENF
     *OMG(20)     ,OMGP(20,2),VARQ(9)    ,TOL(9)     ,SVAR(10)  ,WDC(20)    ,  GENF
     *A(9,9)      ,ACON(9)   ,BCON(9)    ,COTI(9,9)  ,DCON(9)   ,DTP        ,  GENF
     *DTS         ,DT        ,G          ,DPSQ       ,Q         ,QS         ,  GENF
     *R           ,RE        ,MACH       ,PA         ,RO        ,CS         ,  GENF
     *VNU         ,PAR       ,ROR        ,CSR        ,VNR   -   ,SUMSQ      ,  GENF
     *SVSQ        ,TIMEPH    ,TIMES      ,TOP        ,TOS       ,TR(9)      ,  GENF
     *TST(20)     ,TPH   (20),DIS(20)    ,DIP(20)    ,T         ,W          ,  GENF
     *TLP1(20)    ,TLS1  (20),DIP1(20)   ,DIS1(20)   ,TIME      ,OMP        ,  GENF
     *TIMPR       ,LIFT      ,DRAG       ,TAX        ,TBURN     ,TBU(20)    ,  GENF
     *AE          ,FP        ,FPOLD      ,FPD        ,MACHR     ,MACHV      ,  GENF
     *QR          ,QV        ,FVAC       ,LIFTV      ,                         GENF
     *LIFTR       ,LIFTA     ,           ,DRAGV      ,DRAGR     ,DRAGA      ,  GENF
     *            ,LIFTM     ,DBR        ,DB         ,ISP       ,ISPF       ,  GENF
     *                        ULFT       ,ULFTV      ,ULFTR     ,ULFTA      ,  GENF
     *XMCG        ,XMCGV     ,XMCGR      ,XMCGA      ,XMCGM     ,CODAE      ,  GENF
     *CULFT       ,CT        ,CALPHA     ,CDE        ,DELTAE    ,SID        ,  GENF
     *COD         ,SIDAE     ,XCG        ,ZCG        ,XJ                       GENF
      COMMON / GENF  /                                                   GENF
     *XJV         ,XJR       ,GH         ,GAMMAD     ,XKG       ,XKP        ,  GENF
     *FRATED    ,IRATED      ,                                           GENF
     *P1          ,P2        ,P3       ,XK1          ,XK2       ,XK3        ,  GENF
     *XK1T        ,XK2T      ,XK3T       ,XK1D       ,XK2D      ,XK3D       ,  GENF
     *XK1A        ,XK2A      ,XK3A       ,XK1V       ,XK2V      ,XK3V       ,  GENF
     *XK1G        ,XK2G      ,XK3G       ,XK1P       ,XK2P      ,XK3P       ,  GENF
     *XK1R        ,XK2R      ,XK3R       ,XK1D       ,XK2D      ,XK3D       ,  GENF
     *XK1U        ,XK2U      ,XK3U       ,XK1M       ,XK2M      ,XK3M       ,  GENF
     *PV          ,PG        ,PP         ,PR         ,PO        ,DPDY(3,8)     GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                        GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                 GENF
      COMMON/STATE3/                                                     STATE3D
     *VAR(14)     ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)      ,SVY(10)    ,  STATE3D
     *XL(9,9)     ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15)  ,  STATE3D
     *SINPSI      ,COSPSI    ,SINRHO     ,COSRHO     ,OCORHO    ,OCORO2     ,  STATE3D
     *SVBV    (9),OMEGA      ,OMEGA2     ,                                   STATE3D
     *VDV         ,GDV       ,RDV        ,MDV        ,PDV       ,ODV        ,  STATE3D
     *UDV         ,VDG       ,GDG        ,RDG        ,PDG       ,ODG        ,  STATE3D
     *UDG         ,VDR       ,GDR        ,MDR        ,PDR       ,ODR        ,  STATE3D
     *UDR         ,VDM       ,GDM        ,MDM        ,PDM       ,VDP        ,  STATE3D
     *GDP         ,PDP       ,QDP        ,UDP        ,VDO       ,GDO        ,  STATE3D
     *PDO         ,UDO       ,HTDV       ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                STATE3D
      COMMON/STATE3/                                                     STATE3D
     *SIN2RO      ,COS2RO      ,COS2GM                                   STATE3D
      COMMON/PARAM/                                                      PARAM
     *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),            PARAM
     *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                        PARAM
     *,DELP(9)                                                           PARAM
      COMMON /XCODES/                                                    XCODES
     *ITQ     (9),ICOR  (20),ITI         ,INTB       ,JGID(20,2),JPH (20,2),  XCODES
     *JST   (20)             ,NCNST      ,NSB        ,NSAB      ,NICNB      ,  XCODES
     *I2OP        ,ICOP      ,IFAW       ,IFAR       ,IFB       ,INO        ,  XCODES
     *IOPEN       ,IPH       ,ISPH       ,ISST       ,IARC      ,ISTART     ,  XCODES
     *ITCT        ,ITER      ,IVAR       ,JK         ,JPS       ,JS         ,  XCODES
     *KOP         ,KPST      ,K          ,KST        ,NAD       ,NCASE      ,  XCODES
     *NCN         ,NEQB      ,NEQ        ,NOP        ,NPH       ,N          ,  XCODES
     *NST         ,IPST      ,IPRINT     ,ISTN       ,IPHN      ,ISTNB      ,  XCODES
     *IPHNB       ,IBLK1     ,IBLK2      ,ISTOP      ,ISTPP     ,L          ,  XCODES
     *IFOB        ,NB        ,LB         ,MB         ,NPHP      ,NPHB       ,  XCODES
     *NCTIN       ,NEQF  ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3     XCODES
      EQUIVALENCE(NPA,NPAR)                                              ADJUST
      ENTRY ADJUS                                                        ADJUST
C   I      THIS  COMPUTES  ADJUSTMENTS IN  INITIAL CONDITIONS            ADJUST
C   IA   SCAN  THE  POINTER  ARRAY  FOR  INITIAL  STATES  AND  FIRST     ADJUST
      NPAR = NPARA                                                       ADJUST
C          RESET INITIAL STATES                                          ADJUST
      DO 10 I=1,NPARA                                                    ADJUST
      IF(IPOINT(I).LE.21) GO TO 10                                       ADJUST  10
      KK=IPOINT(I)-21                                                    ADJUST
      VAR(KK)=PARA(I)                                                    ADJUST
```

```
      SVAR(KK+1)=VAR(KK)                                              ADJUST
   10 CONTINUE                                                        ADJUST
C                                                                     COMN
C       I-B   SAVE  SENSITIVITIES  FOR  USE IN  MTX                   COMN
C                                                                     COMN
      DO 20 I=1,NCN                                                   ADJUST
      DO 20 J=1,NPARA                                                 ADJUST
   20 SPARB(I,J)= SPARA(I,J)                                          ADJUST
C         ARC  TIME  DURATION                                         ADJUST
      ITI  = 0                                                        ADJUST
      KM=0                                                            ADJUST
   30 CONTINUE                                                        ADJUST
      CALL MTX                                                        ADJUST
      CALL MTX1                                                       ADJUST
C                                                                     COMN
C       I-C  COMPUTE  INITIAL STATE  CORRECTIONS                      COMN
C              AND REMOVE SENSITIVITY  FROM  MATRIX                   COMN
      DO 110  I=1,NPARA                                               ADJUST
      IF(IPOINT(I).EQ 1)  ITI=I                                       ADJUST
      IF(IPOINT(I).LE 21) GO TO 110                                   ADJUST
      KK=  IPOINT(I) - 21                                             ADJUST
      VAR(KK) = PARA(I) + DPAR(I)                                     ADJUST
      SVAR(KK+1)= VAR(KK)                                             ADJUST
      NPAR = NPAR-1                                                   ADJUST
      DO 100  J=1,NCN                                                 ADJUST
  100 SPARB(J,I)=0                                                    ADJUST
  110 CONTINUE                                                        ADJUST
      DO 120 J=1,NCN                                                  ADJUST
      DO 120 I=1,NCN                                                  ADJUST
  120 S2INV(I,J)=0                                                    ADJUST
      RETURN                                                          ADJUST
C                                                                     COMN
C          ENTRY  TO  ELIMINATE  ARC TIME SENS. FROM  MATRIX          COMN
C                                                                     COMN
      ENTRY  ADJIN                                                    ADJUST
C     IV   TEST FOR  PREVIOUS ARC  STAGING                            ADJUST
      IF(ITI  EQ 0) GO TO 150                                         ADJUST
      NPAR= NPAR-1                                                    ADJUST
      DO 130 I=1,NCN                                                  ADJUST
  130 SPARB(I,ITI )= 0                                                ADJUST
      ITI=0                                                           ADJUST
      DO 140 I=1,NCN                                                  ADJUST
      DO 140 J=1,NCN                                                  ADJUST
  140 S2INV(I,J)=0.                                                   ADJUST
      IF(NPAR EQ 0) RETURN                                            ADJUST
C      III   SCAN  FOR  ARC  TIME  ADJUSTMENT                         ADJUST
      ITI  = 0                                                        ADJUST
  150 DO 160 I=1,NPARA                                                ADJUST
      IF(IPOINT(I).EQ IARC) ITI =I                                    ADJUST
  160 CONTINUE                                                        ADJUST
      RETURN                                                          ADJUST
      ENTRY ADJUT                                                     ADJUST
C     IV    ADJUST  STAGING  TIME                                     ADJUST
C     IVA   TEST  FOR  WHETHER ARC TERMINATION HAS STARTED            ADJUST
      IF(KST.GT 0) RETURN                                             ADJUST
      OMG(IARC) = PARA(ITI ) + DPAR(ITI )                             ADJUST
C     IVB   TEST FOR  CONCURRENT PHASING                              ADJUST
      IF(ICOR(IPH) GT.IARC) RETURN                                    ADJUST
      OMGP(IPH,2) =OMG(IARC)                                          ADJUST
      OMP= OMG(IARC)                                                  ADJUST
      RETURN                                                          ADJUST
      END                                                             ADJUST
```

SUBRØUTINE AGETB3

Subroutine AGETB3

Purpose

Fetches state and control from random access file for use in adjoint solution.

Description

This routine has two entry points.

Entry BEGNA: This entry initializes reading the forward trajectory at final trajectory time. This initialization involves determining the last buffer number that was written into and how many words were written there. Starting from there the program may proceed backwards to extract data at monotonically decreasing time points to correspond exactly with the adjoint integration time points. BEGNA is called from BNTG.

Entry AGETB: This entry simply steps backward in the buffer array and loads the trajectory data into the appropriate common locations. When the starting location of the buffer is reached, the next lower buffer is loaded in from the random file and the process of backward data extraction starts over. AGETB is called from MODELB.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
APHR	α	O	Angle of attack (DEG)	/AECO3 /(	2)	AGETB3 AST3 BEROCO BLGCON GUI3A MODELA MODELB MTX3A OUT	O M I O M M I O I	APHR APHR APHR APHR APHR APHR APHR APHR APHR
E		I	Storage retrieval buffer	/AGETB3/(*	)	AGETB3	I	E
FTIME		O	Time at which trajectory data set is stored. (SEC)	/RETREV/(	1)	AGETB3 AST3 GETIT TRAN3	O O I I	FTIME FTIME FTIME FTIME
IBLK2		M	Storage retrieval buffer counter	/XCODES/(	174)	AGETB3 AST3	M M	IBLK2 IBLK2
IBUF2		M	Counts number of buffers as forward trajectory data is retrieved from random file.	/RETREV/(	9)	AGETB3 AST3	M M	IBUF2 IBUF2
IFAR		I	File where nominal trajectory data is read from	/XCODES/(	139)	AGETB3 AST3 TOPM	I I M	IFAR IFAR IFAR
IPHN		M	Stored history data phase number	/XCODES/(	170)	AGETB3 AST3 GETIT	M M I	IPHN IPHN IPHN
ISTN		O	Stored history data arc number	/XCODES/(	169)	AGETB3 AST3 GETIT TRAN3	O O I I	ISTN ISTN ISTN ISTN
JP1		M	Option flag for first governing equation	/XCODES/(	217)	AGETB3 AST3 MODELA MODELB PROPB PROPIN	M M M I O O	JP1 JP1 JP1 JP1 JP1 JP1
JP3		O	Option flag for third governing equation	/XCODES/(	219)	AGETB3 AST3 MODELA MODELB OUT PROPIN	O M M I I O	JP3 JP3 JP3 JP3 JP3 JP3
MAXA		I	Number of words in last stored partial buffer of trajectory data MAXA(1) corresponds to random file 39 MAXA(2) corresponds to random file 40.	/RETREV/(	3)	AGETB3 AST3	I M	MAXA MAXA
MIXA		I	Maximum number of words in trajectory data buffer = 990.	/RETREV/(	12)	AGETB3 AST3 SDINP TOPM	I I I O	MIXA MIXA MIXA MIXA
MXA		I	Index of last stored word in full buffer of forward trajectory data.	/RETREV/(	14)	AGETB3 AST3 SDINP	I I O	MXA MXA MXA
NBUFA		I	Number of buffers of trajectory data stored on random files 39 and 40 respectively.	/RETREV/(	6)	AGETB3 AST3	I M	NBUFA NBUFA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NEQ		I	Number of integrated states		/XCODES/(	162)	ADICB3	I	NEQ
							ADIC3A	I	NEQ
							ADID3A	I	NEQ
							AGETB3	I	NEQ
							AST3	I	NEQ
							BGET3	I	NEQ
							BSTO3	I	NEQ
							MTX3A	I	NEQ
							OUT	I	NEQ
							REU3	I	NEQ
							SDER3	I	NEQ
							SDINP	M	NEQ
							TOPM	I	NEQ
							TRAN3	I	NEQ
							YREF3	I	NEQ
PHID	ϕ	O	Bank angle	(DEG)	/AECO3 /(	10)	AGETB3	O	PHID
							AST3	M	PHID
							GUI3A	M	PHID
							MODELA	M	PHID
							MODELB	I	PHID
							MTX3A	O	PHID
							OUT	I	PHID
VAR	v	O	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	Y
							STP3	I	VAR
							TOPM	D	KWOW
							YREF3	M	V

```
      SUBROUTINE AGETB3                                                AGETB3
C                                                                      COMN
C           FETCHES  STATE AND CONTROL FROM RANDOM ACCESS FILE         COMN
C           FOR  USE  IN ADJOINT  SOLUTION                             COMN
C           ENTRY  BEGNA  INITIALIZES  BUFFERS                         COMN
C           ENTRY  AGETB  FETCHES DATA                                 COMN
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                          RETREV
      COMMON/RETREV/                                                   RETREV
     *NBUFA(2) ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA     , RETREV
     *MIXB     ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB      RETREV
     *,NBUFB   ,IBUFB                                                  RETREV
      COMMON/STATE3/                                                   STATE3D
     *VAR(14)  ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)  ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI   ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA    ,OMEGA2    ,                                 STATE3D
     *VDV      ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE3D
     *UDV      ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE3D
     *UDG      ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE3D
     *UDR      ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP      ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE3D
     *PDO      ,UDO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                              STATE3D
      COMMON/STATE3/                                                   STATE3D
     *SIN2RO   ,COS2RO    ,COS2GM                                      STATE3D
      COMMON/AEC03/                                                    AEC03
     *APHO     ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      , AEC03
     *SINA     ,COSA      ,PHID      ,PHID      ,PHI       ,SINPHI   , AEC03
     *COSPHI   ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDD      , AEC03
     *CDOM     ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM     , AEC03
     *CM       ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM      , AEC03
     *CLAM     ,CL        ,CLA       ,CLM       ,                      AEC03
     *CD       ,CDA       ,CDM                                         AEC03
      COMMON /XCODES/                                                  XCODES
     *ITQ    (9),ICOR  (20),ITI      ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)          ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP     ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN    ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT     ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP      ,KPST      ,K         ,KST       ,NAD       ,NCASE    , XCODES
     *NCN      ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST      ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB    ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFDB     ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN    ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3       XCODES
      COMMON/GENF/                                                     GENF
     *OMG(20)  ,OMGP(20,2),VARO(9)   ,TOL(9)    ,SVAR(10)  ,WOC(20)  , GENF
     *A(9,9)   ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS      ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R        ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU      ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ     ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)  ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20) ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR    ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE       ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR       ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR    ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *           LIFTM    ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                     ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG     ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT    ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD      ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                 GENF
     *XJV      ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED   ,IRATED    ,                                            GENF
     *P1       ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T     ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A     ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G     ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R     ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U     ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV       ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,              MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                 FRAT
```

```
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      DIMENSION  E(990)                                                 AGETB3
C          I     INITIALIZE BUFFER NUMBER  AND READ FILE                COMN
      ENTRY BEGNA                                                       AGETB3
      KK= NEQ+1                                                         AGETB3
      IBUF2 =NBUFA(IFAR-38) +1                                          AGETB3
      IBLK2 = MAXA(IFAR-38) +1                                          AGETB3
      GO TO 20                                                          AGETB3
   10 IF(IBUF2.LE. 1) RETURN                                            AGETB3
      IBLK2 = MXA +1                                                    AGETB3
   20 IBUF2 = IBUF2 -1                                                  AGETB3
      CALL READMS(IFAR,E,MIXA,IBUF2)                                    AGETB3
      RETURN                                                            AGETB3
      ENTRY AGETB                                                       AGETB3
C                                                                       COMN
C          II     EXTRACT  DATA FROM  BUFFER                            COMN
      IBLK2 = IBLK2-1                                                   AGETB3
      PHID= E(IBLK2)                                                    AGETB3
      IBLK2= IBLK2 -1                                                   AGETB3
      APHR = E(IBLK2)                                                   AGETB3
      DO 30  I=1,NEQ                                                    AGETB3
      IBLK2 = IBLK2-1                                                   AGETB3
      JJ = KK-I                                                         AGETB3
   30 VAR( JJ) = E(IBLK2)                                               AGETB3
      FTIME  = E(IBLK2-1)                                               AGETB3
      IPHN=E(IBLK2-2)/10000                                             P014
      JP1 = (E(IBLK2-2)-10000*IPHN)/100                                 P014
      JP3= E(IBLK2-2) - 10000*IPHN - 100*JP1 +.2                        P014
      ISTN   = E(IBLK2-3)                                               AGETB3
      IBLK2  = IBLK2-3                                                  AGETB3
      IF(IBLK2.LE.1) GO TO 10                                           AGETB3
C                                                                       COMN
C          III   IF BUFFER  EMPTY  GO UP TO 20  AND FILL BUFFER         COMN
      RETURN                                                            AGETB3
      END                                                               AGETB3
```

20 OCT 72 6.01-46

SUBRØUTINE
ANLATM

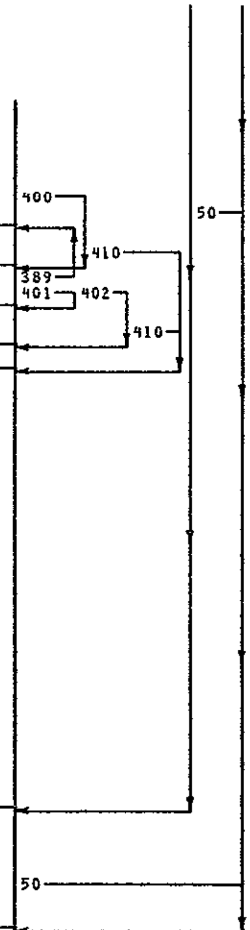
```
      SUBROUTINE ANLATM(H,DIM,INDP)                                     ANLAT
      DIMENSION  DIM(8)                                                 ANLAT
      COMMON/GLOBAL/IDUMM(91),JP3                                       APR27
      COMMON/SPECO/  UMU2,RORR                                          ANLAT
      DATA A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17,  ANLAT
     *A18,A19,A20,A21,A22,A23,A24,A25,A26,A27,A28,A29,A30,A31,A32       ANLAT
     *   / -.14655396E-6,6356 77,.25653341E-10,6356.77,.14116834E-3,    ANLAT
     *   14.002385,- 38282910E-4,216 23225, 15084978E-3,26.414270,      ANLAT
     *   684.10967,.67419880E-3, 044294588,-58500460,.85519675E-4,      ANLAT
     *   137.47450,10533.544,.49863416E-4,.013120767,.90188546,         ANLAT
     *  -.25392354E-3,-193 32352,10180.367,.11921879E-2,.034567717,     ANLAT
     *  -3.3413764,-.33888604E-4,-384.32662,38131 516,.89812379E-4,     ANLAT
     *   .028810210,-5.5362354/                                         ANLAT
      DATA B9,B8,B7,B6,B5,B4,B3,B2,B1,B0                                ANLAT
     */  12840084E  2,- 25307047E  0, .22460528E- 2,-.11672987E- 4,     ANLAT
     *  .38922542E- 7,- 86034597E-10, .12577373E-12,-.11709468E-15,     ANLAT
     *    62944405E-19,- 14878777E-22/                                  ANLAT
      DATA  CN/ -3483.67635/                                            ANLAT
      DATA C10,C9,C8,C7,C6,C5,C4,C3,C2,C1,C0                            ANLAT
     *   / 283 7492391, -3.955242007,- 5232974573, .05256403630,        ANLAT
     *   - 1832962145E-2,  3432295909E-4, - 3930824139E-6,  2848535349E ANLAT
     *   -8, -.1269919974E-10, .3161762924E-13, -.3350145769E-16/       ANLAT
      DATA D3,D2,D1,D0                                                  ANLAT
     *   / 253.316666, 6 8049727, -.56317016E-2,  .12742813E-5 /        ANLAT
      DATA E9,E8,E7,E6,E5,E4,E3,E2,E1,E0                                ANLAT
     */  47255200E  2,-.96148320E  0, .87302363E- 2,-.96139476E- 4,     ANLAT
     *   15580020E- 6,-.34772427E- 9, .51217501E-12,-.47967392E-15,     ANLAT
     *  .25907821E-18,- 61476982E-22/                                   ANLAT
      DATA      CC8,CC7,CC6,CC5,CC4,CC3,CC2,CC1,CC0                     ANLAT
     *   /               -1 0465949146,.1576921089,-.733184858E-2,      ANLAT
     *      1.7161479545E-4,-2.3584944834E-6,1 9939747443E-8,           ANLAT
     *     -1.0159359792E-10,2 8455866316E-13,-3.350145769E-16/         ANLAT
      DATA       CCC7,CCC6,CCC5,CCC4,CCC3,CCC2,CCC1,CCC0                ANLAT
     *   /                  3153842178,-2 1995545 74E-2,6.864591818E-4, ANLAT
     *     -1.1792472417E-5,1.1963848 4658E-7,-7.1115518544E-10,        ANLAT
     *      2 27646930528E-12,-3.0151311921E-15/                        ANLAT
      DATA ROWXTO /3.5298375E11/                                        ANLAT
      DATA       DD1,DD0                                                ANLAT
     *   /             -1.12634032E-2,.38228439E-5/                     ANLAT
      DATA       DDD0                                                   ANLAT
     *   /                .76456878E-5/                                 ANLAT
      DATA      BB7,BB6,BB5,BB4,BB3,BB2,BB1,BB0                         ANLAT
     *   /              44921056E-2,- 35018961E-4,1.55690168E-7,        ANLAT
     *     -4.30172985E-10,.75464238E-12,-.81966276E-15,5.0355524E-19,  ANLAT
     *     -1.33908993E-22/                                             ANLAT
      DATA      EE7,EE6,EE5,EE4,EE3,EE2,EE1,EE0                         ANLAT
     *   /              1.74604726E-2,-1 38418428E-4,.6232008E-6,       ANLAT
     *     -1.73862135E-9,3.07305006E-12,-3.35771744E-15,2.07262568E-18,ANLAT
     *     -5.53292838E-22/                                             ANLAT
      DATA       EEE6,EEE5,EEE4,EEE3,EEE2,EEE1,EEE0                     ANLAT
     *   /                 -2.76836856E-4,1 8696024E-6,-6.9544854E-9,   ANLAT
     *      1.53652503E-11,-2 014630464E-14,1.450837976E-17,            ANLAT
     *     -4.426342704E-21/                                            ANLAT
      DATA CON, HAFCON                                                  ANLAT
     */2.0046802776378 6E-2,1.00234013881893E-2/                        ANLAT
      DATA POWERO, P0/-.28016066E-2, 101325E9/                          ANLAT
      DATA X2,X3,X4,X5,X6,X7,X9,X11,X12                                 ANLAT
     */2.0885400839168 93E-5,1.9403172295773 53E-12,3 280839895013123E3,ANLAT
     *5 4864E-4,6.3658701757786 57E-9,5 9140869157517 31E-16,           ANLAT
     *1.67225472E-7,1 8026136919211 16E-19,3.048E-4/                    ANLAT
      DATA  COEF1,SC,TF,UMUF,UMUF2,UMUF3,TFSC/                          ANLAT
     *      1.7818E-4, 198 , 392., 3 02E-7, 9.1204E-14, 27.543608E-21,  ANLAT
     * 590./                                                            ANLAT
      POWER(Z) = A1/(Z + A2) + A3*ALOG(Z + A4) + A5*ALOG(Z + A6) +      ANLAT
     *               A7*ALOG(A8 - Z) + A9*ALOG(Z*(Z - A10) + A11) +     ANLAT
     *               A12*ATAN(A13*Z - A14) + A15*ALOG(Z*(Z - A16) + A17) +ANLAT
     *               A18*ATAN(A19*Z - A20) + A21*ALOG(Z*(Z + A22) + A23) +ANLAT
     *               A24*ATAN(A25*Z + A26) + A27*ALOG(Z*(Z + A28) + A29) +ANLAT
     *               A30*ATAN(A31*Z + A32)                              ANLAT
C                                                                       ANLAT
   40 X =  H * X12                                                      ANLAT
      IF (X .GT. 550.)  X=550.                                          ANLAT
      IF (X .LT. 0.   )  X= 0.                                          ANLAT
      IF ( X.GT  195. )   GO TO 1000                                    ANLAT   1000
C                                                                       ANLAT
```

20 OCT 72 6.01-46

```
C      SET   CONSTANTS                                                  ANLAT
       POWERX= POWER(X)                                                 ANLAT
C                                                                       ANLAT
CTM                                                                     ANLAT
       TM = C10 + X*(C9 + X*(C8 + X*(C7 + X*(C6 + X*(C5 + X*(C4 + X*(C3 + ANLAT
     *      X*(C2 + X*(C1 + X*C0))))))))))                              ANLAT
       SQRTTM =SQRT(TM)                                                 ANLAT
       CS = CON*SQRTTM                                                  ANLAT
       EXPO=  EXP( CN*(POWERX -  POWERO) )                              ANLAT
C PRESSURE                                                              ANLAT
       P    =  PO* EXPO                                                 ANLAT
C DENSITY                                                               ANLAT
       RHO =  ROWXTG*EXPO/TM                                            ANLAT
C            DYNAMIC VISCOSITY CALCULATIONS                             ANLAT
C                                                                       ANLAT
     7 TTF32  = (1.8*TM/TF)*SQRT(1 8*TM/TF)                             ANLAT
       TSC    = 1 8*TM + SC                                             ANLAT
       UMU    = COEF1* TTF32/TSC                                        ANLAT
       IF(INDP.GT.0)  GO TO 500                                         ANLAT
       IF(JP3.LT 7)  GO TO 50                                           ANLAT
   499 ASSIGN 600 TO IGO                                                ANLAT
       GO TO 510                                                        ANLAT
   500 IF(JP3.LT.7) GO TO  499                                          ANLAT
       TM2 =  CC8 + X*(CCC7 + X*(CCC6 + X*(CCC5 + X*( CCC4 + X*( CCC3 +   ANLAT
     *        X*(CCC2 + X*(CCC1 + X*CCC0)))))))                         ANLAT
       IF(JP3-8) 501, 501 ,502                                          ANLAT
   501 ASSIGN 601 TO IGO                                                ANLAT
       GO TO 510                                                        ANLAT
   502 ASSIGN 602 TO IGO                                                ANLAT
   510 CONTINUE                                                         ANLAT
       TM1 =  C9 + X*(CC8 + X*(CC7 + X*(CC6 + X*(CC5 + X*( CC4 + X*(CC3 + ANLAT
     *        X*(CC2 + X*(CC1 + X*CC0))))))))                           ANLAT
       CS1= HAFCON * TM1 /SQRTTM                                        ANLAT
       R    =  6356.765 + X                                             ANLAT
       G    =  .396271577E6/R**2                                        ANLAT
       GTM =    G/TM                                                    ANLAT
       EXP1=  EXPO*CN*GTM                                               ANLAT
       P1   =  PO* EXP1                                                 ANLAT
       RHO1= (ROWXTO*EXP1 --RHO*TM1)/TM                                 ANLAT
       TMS = 1.8 *TM                                                    ANLAT
       TMS1= X5* TM1                                                    ANLAT
       GT   = TFSC/ TSC                                                 ANLAT
       GT1 = -GT/TSC                                                    ANLAT
       FT   = TTF32                                                     ANLAT
       FT1 = 1.5* FT /TMS                                               ANLAT
       UMU1 = UMUF *(FT*GT1*TMS1 + GT *FT1*TMS1)                        ANLAT
       GO TO  IGO                                                       ANLAT
   602 CONTINUE                                                         ANLAT
       TMS2 = X9*TM2                                                    ANLAT
       GT2 = 2.* GT /(TSC*TSC)                                          ANLAT
       TM20 = TMS*TMS                                                   ANLAT
       FT2= .75 *FT /TM20                                               ANLAT
       UMU2 = UMUF*((FT*GT1 +GT*FT1)*TMS2 +(FT*GT2 +2.*FT1*GT1 +GT*FT2)  ANLAT
     *     *TMS1*TMS1   )                                               ANLAT
   601 CONTINUE                                                         ANLAT
       G1=-2.*G/R                                                       ANLAT
       GTM1=(G1*TM-G*TM1)/(TM*TM)                                       ANLAT
       EXP2= CN*(EXP1*GTM+EXPO*GTM1)                                    ANLAT
       RHO2= (ROWXTO*EXP2-2.*RHO1*TM1-RHO*TM2)/TM                       ANLAT
       RORR = X11*RHO2                                                  ANLAT
       GO TO 600                                                        ANLAT
  1000 CONTINUE                                                         ANLAT
C                                                                       ANLAT
C  TM .GT  195                                                          ANLAT
       TM = D3 + X*(D2 + X*(D1 + X*D0))                                 ANLAT
       SQRTTM =SQRT(TM)                                                 ANLAT
       CS=CON*SQRTTM                                                    ANLAT
       TTF32=(1.8*TM/TF)*SQRT(1.8*TM/TF)                                UH
       TSC  =1.8*TM+SC                                                  UH
       UMU=COEF1*TTF32/TSC                                              UH
C                                                                       ANLAT
C ROW .GT. 195                                                          ANLAT
```

500
510
499
501
502
510
600
50

```
C   P   GT  195                                                        ANLAT
      RHO = E9 + X*(E8 + X*(E7 + X*(E6 + X*(E5 + X*(E4 + X*(E3 + X*(E2 + ANLAT
     *        X*(E1 + X*E0))))))))                                      ANLAT
 \    P = B9 + X*(B8 + X*(B7 + X*(B6 + X*(B5 + X*(B4 + X*(B3 + X*(B2 +   ANLAT
     *     X*(B1 + X*B0))))))))                                         ANLAT
      IF(INDP GT 0) GO TO 400                                           ANLAT
      IF(JP3 LT 7)GO TO 50                                              ANLAT
  389 ASSIGN 600 TO IGO                                                 ANLAT
      GO TO 410                                                         ANLAT
  400 IF(JP3 LT.7) GO TO 389                                            ANLAT
      IF(JP3-8) 401,401,402                                             ANLAT
  401 ASSIGN 701 TO IGO                                                 ANLAT
      GO TO 410                                                         ANLAT
  402 ASSIGN 702 TO IGO                                                 ANLAT
  410 CONTINUE                                                          ANLAT
      RHO1 =  E8 + X*(EE7 + X*(EE6 + X*(EE5 + X*(EE4 + X*(EE3 + X*(EE2 + ANLAT
     *         X*(EE1 + X*EE0)))))))                                    ANLAT
      P1 =  B8 + X*(BB7 + X*(BB6 + X*(BB5 + X*(BB4 + X*(BB3 + X*(BB2 +  ANLAT
     *       X*(BB1 + X*BB0)))))))                                      ANLAT
      TM1 =  D2 + X*(DD1 + X*DD0)                                       ANLAT
      CS1=HAFCON*TM1/SQRTTM                                             UH
      TMS=1.8*TM                                                        ANLAT
      TMS1=X5*TM1                                                       ANLAT
      GT  = TFSC/TSC                                                    ANLAT
      GT1 = -GT/ TSC                                                    ANLAT
      FT  = TTF32                                                       ANLAT
      FT1= 1.5 *FT / TMS                                                ANLAT
      UMU1 = UMUF *(FT*GT1*TMS1 + GT *FT1*TMS1)                         ANLAT
      GO TO IGO                                                         ANLAT
  702 CONTINUE                                                          ANLAT
      TM2 =  DD1 + X*DDD0                                               ANLAT
      TMS2 = X9*TM2                                                     ANLAT
      GT2 = 2 *GT /(TSC*TSC)                                            ANLAT
      TM20= TMS*TMS                                                     ANLAT
      FT2 = .75*FT /TM20                                                ANLAT
      UMU2= UMUF *((FT*GT1 + GT*FT1) *TMS2 +(FT*GT2 +2 *FT1*GT1 +GT*FT2) ANLAT
     *    *TMS1*TMS1    )                                               ANLAT
  701 CONTINUE                                                          ANLAT
      RHO2 =  EE7 + X*(EEE6 + X*(EEE5 + X*(EEE4 + X*(EEE3 + X*(EEE2 +   ANLAT
     *         X*(EEE1 + X*EEE0))))))                                   ANLAT
      RORR=X11*RHO2                                                     ANLAT
  600 DIM(5)=X6*P1                                                      ANLAT
      DIM(6)=X7*RHO1                                                    ANLAT
      DIM(7)=CS1                                                        ANLAT
      DIM(8)=UMU1                                                       ANLAT
      IF(X.GT.0.) GO TO 50                                              PHISZ
      DO 605 III=5,8                                                    PHISZ
  605 DIM(III)=0.                                                       PHISZ
   50 DIM(1)=X2*P                                                       ANLAT
      DIM(2)=X3*RHO                                                     ANLAT
      DIM(3)=X4*CS                                                      ANLAT
      DIM(4)=UMU                                                        ANLAT
      RETURN                                                            ANLAT
      END                                                               ANLAT
```



20 OCT 72 G 01-46

SUBRØUTINE AST3

Subroutine AST3

Purpose

This routine handles both storage and retrieval of trajectory data during forward trajectory integration. Each entry point is defined herein.

Description

Entry BEGWR: Initializes counters for storage of trajectory on random file. BEGWR is called from FNTG.

Entry ENDWR: Saves values of counters after trajectory has been stored and "dumps" last partial buffer on random file. ENDWR is called from FNTG.

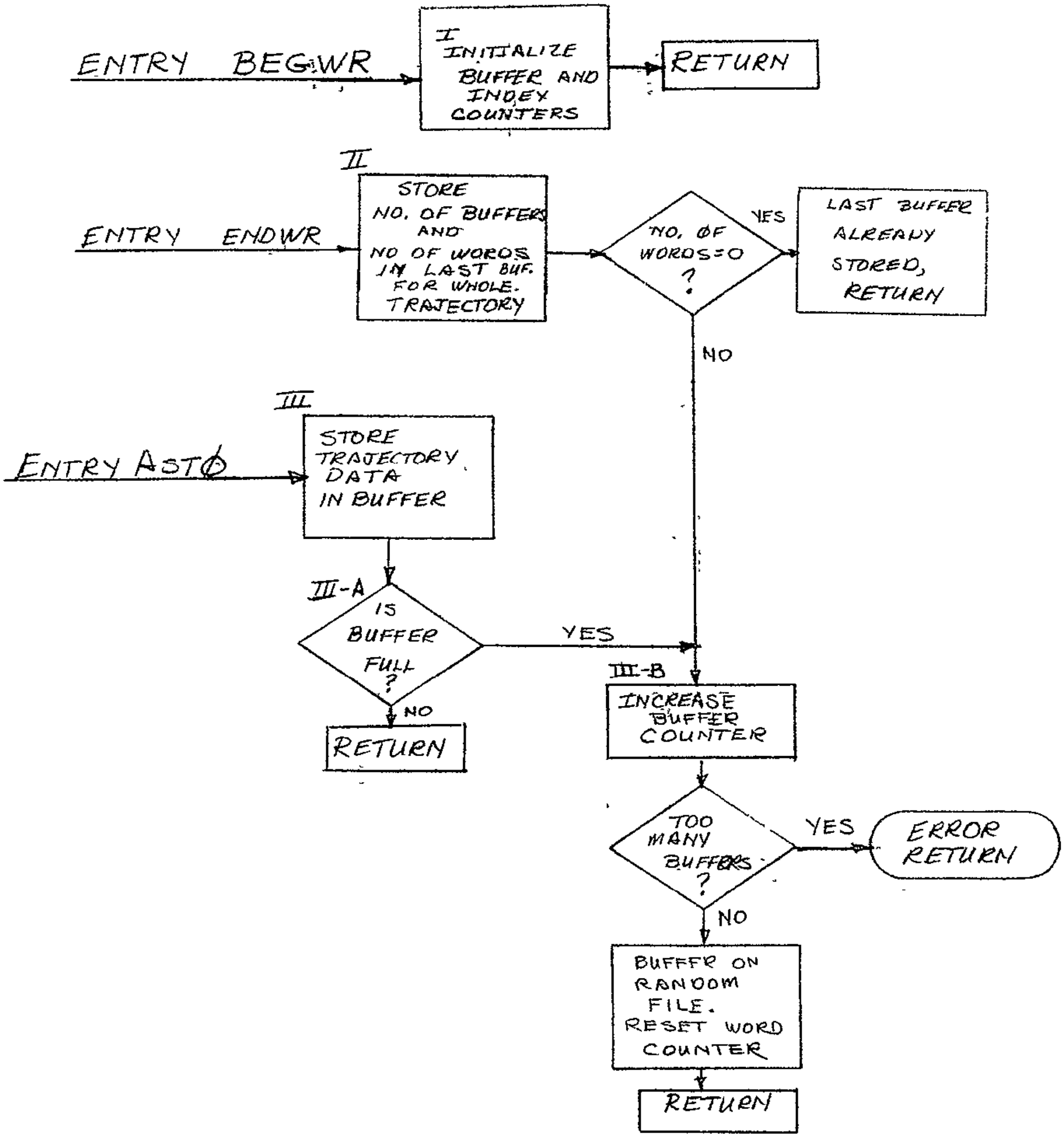
Entry ASTØ: Stores trajectory data in buffer and when buffer is full "dumps" buffer on random file. ASTØ is called from entry CORVAR in REU3 and also FNTG.

Entry BGRA: Initializes read of trajectory data.

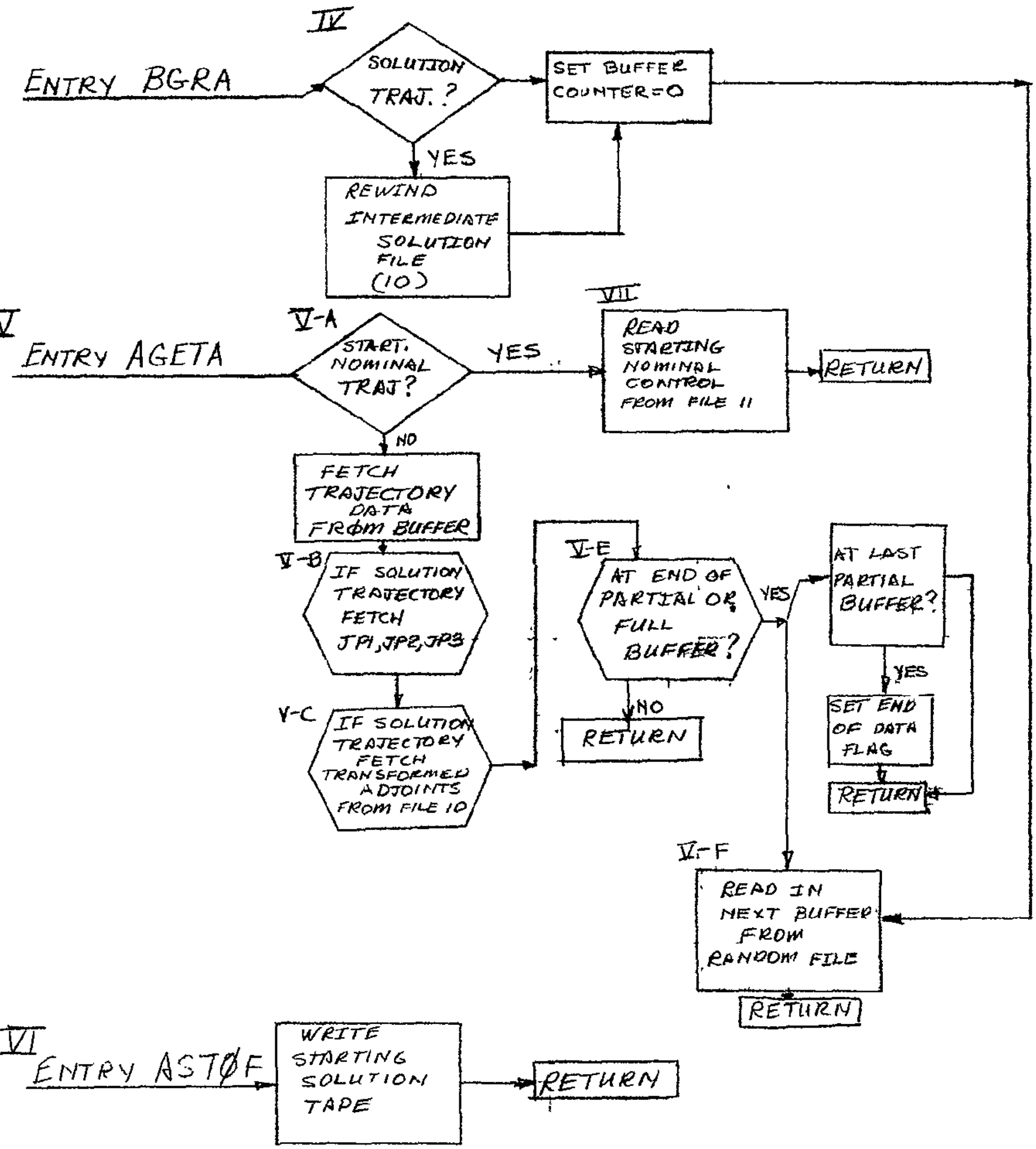
Entry AGETA: Reads data from storage buffer, or if on a starting trajectory, reads data from starting solution file 11. AGETA is called from MODELA.

Entry ASTØF Writes starting solution on file 11. ASTOF is called from FNTG.

SUBROUTINE AST3



SUBROUTINE AST3 (CONTINUED)



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
APHO	α_{old}	M	Angle of attack from last nominal trajectory (DEG)	/AECO3 /(	1)	AST3	M	APHO
						FNTG	I	APHO
						MTX3A	I	APHO
						OUT	I	APHO
						PROPB	O	AEZRO
						PROPIN	O	AEZRO
APHR	α	M	Angle of attack (DEG)	/AECO3 /(	2)	AGETB3	O	APHR
						AST3	M	APHR
						BEROCO	I	APHR
						BLGCON	O	APHR
						GUI3A	M	APHR
						MODELA	M	APHR
						MODELB	I	APHR
						MTX3A	O	APHR
						OUT	I	APHR
D		W	Storage buffer	/AST3 /(*	)	AST3	W	D
E		I	Storage retrieval buffer	/AST3 /(*	)	AST3	I	E
FTIME		O	Time at which trajectory data set is stored (SEC)	/RETREV/(	1)	AGETB3	O	FTIME
						AST3	O	FTIME
						GETIT	I	FTIME
						TRAN3	I	FTIME
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
IBLK1		M	Storage retrieval buffer counter	/XCODES/(	173)	AST3	M	IBLK1
						BSTO3	M	IBLK1
IBLK2		M	Storage retrieval buffer counter	/XCODES/(	174)	AGETB3	M	IBLK2
						AST3	M	IBLK2
IBUF1		M	Counts number of buffers as forward trajectory is stored on random file.	/RETREV/(	8)	AST3	M	IBUF1
IBUF2		M	Counts number of buffers as forward trajectory data is retrieved from random file.	/RETREV/(	9)	AGETB3	M	IBUF2
						AST3	M	IBUF2
IFAR		I	File where nominal trajectory data is read from.	/XCODES/(	139)	AGETB3	I	IFAR
						AST3	I	IFAR
						TOPM	M	IFAR
IFAW		I	Logical file to write current trial trajectory data	/XCODES/(	138)	AST3	I	IFAW
						TOPM	M	IFAW
IND		I	Flag indicates whether on first nominal trajectory (IND=1)	/XCODES/(	141)	AST3	I	IND
						BGET3	I	IND
						FNTG	M	IND
						GUI3A	I	IND
						MTX3A	I	IND
						PROPIN	I	IND
IPH		I	Phase number	/XCODES/(	143)	ADID3A	I	IPH
						ADJUST	I	IPH
						AST3	I	IPH
						BNTG	M	IPH
						FNTG	M	IPH
						GETIT	I	IPH
						GUI3A	I	IPH
						SDINP	M	IPH

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
IPHN		M	Stored history data phase number	/XCODES/(	170)	AGETB3	M	IPHN
						AST3	M	IPHN
						GETIT	I	IPHN
IPST		I	Phase counter for first nominal trajectory	/XCODES/(	167)	AST3	I	IPST
						FNTG	M	IPST
						GUI3A	I	IPST
						REU3	I	IPST
ISTART		O	Initialization and divergance flag	/XCODES/(	147)	AST3	O	ISTART
						BLGCON	O	ISTART
						BLYNE	O	ISTART
						FNTG	I	ISTART
						MODELA	O	ISTART
						PROPIN	O	ISTART
						REU3	I	ISTART
						TEST	M	ISTART
						TOPM	M	ISTART
ISTN		O	Stored history data arc number	/XCODES/(	169)	AGETB3	O	ISTN
						AST3	O	ISTN
						GETIT	I	ISTN
						TRAN3	I	ISTN
ITER		I	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
						FNTG	I	ITER
						GETIT	I	ITER
						MODELA	I	ITER
						OUT	I	ITER
						PAYO2	M	ITER
						PROPIN	I	ITER
						TEST	M	ITER
						TOPM	M	ITER
JP1		M	Option flag for first governing equation	/XCODES/(	217)	AGETB3	M	JP1
						AST3	M	JP1
						MODELA	M	JP1
						MODELB	I	JP1
						PROPB	O	JP1
						PROPIN	O	JP1
JP3		M	Option flag for third governing equation	/XCODES/(	219)	AGETB3	O	JP3
						AST3	M	JP3
						MODELA	M	JP3
						MODELB	I	JP3
						OUT	I	JP3
						PROPIN	O	JP3
K		O	Storage retrieval flag indicates end of arc, phase, or data.	/XCODES/(	156)	AST3	O	K
						FNTG	O	K
						GETIT	M	K
						MODELA	I	K
						SDINP	M	K
LUM		I	Program control flag. LUM = 0· Steepest descent only; LUM = 1 Steepest descent and adjoint transformation stored on tape; LUM = 2: Steepest descent and QL; LUM = 3: QL only	/GLOBAL/(	6)	AST3	I	LUM
						FNTG	I	LUM
						GEINP	I	LUM
						PADS1	M	LUM
						SDINP	I	LUM
						TOPM	M	LUM
MAXA		M	Number of words in last stored partial buffer of trajectory data MAXA(1) corresponds to random file 39 MAXA(2) corresponds to random file 40.	/RETREV/(	3)	AGETB3	I	MAXA
						AST3	M	MAXA
MIXA		I	Maximum number of words in trajectory data buffer = 990	/RETREV/(	12)	AGETB3	I	MIXA
						AST3	I	MIXA
						SDINP	I	MIXA
						TOPM	O	MIXA
MXA		I	Index of last stored word in full buffer of forward trajectory data.	/RETREV/(	14)	AGETB3	I	MXA
						AST3	I	MXA
						SDINP	O	MXA
NBFA		I	Maximum number of buffers permitted to store forward trajectory data = 20	/RETREV/(	10)	AST3	I	NBFA
						TOPM	O	NBFA
NBUFA		M	Number of buffers of trajectory data stored on random files 39 and 40 respectively.	/RETREV/(	6)	AGETB3	I	NBUFA
						AST3	M	NBUFA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NCN		M	Number of elements in $d\Psi$		/XCODES/(	160)	ADEQ3A	I	NCN
							ADICB3	I	NCN
							ADIC3A	I	NCN
							ADID3A	I	NCN
							ADJUST	I	NCN
							AST3	M	NCN
							BNTG	I	NCN
							BST03	I	NCN
							MTX3A	I	NCN
							OUT	I	NCN
							PAY02	M	NCN
							TEST	M	NCN
							TOPM	I	NCN
							TRAN3	I	NCN
							TRTOSZ	I	NCN
NEQ		I	Number of integrated states		/XCODES/(	162)	ADICB3	I	NEQ
							ADIC3A	I	NEQ
							ADID3A	I	NEQ
							AGETB3	I	NEQ
							AST3	I	NEQ
							BGET3	I	NEQ
							BST03	I	NEQ
							MTX3A	I	NEQ
							OUT	I	NEQ
							REU3	I	NEQ
							SDER3	I	NEQ
							SDINP	M	NEQ
							TOPM	I	NEQ
							TRAN3	I	NEQ
							YREF3	I	NEQ
PHID	ϕ	M	Bank angle	(DEG)	/AEC03 /(	10)	AGETB3	O	PHID
							AST3	M	PHID
							GUI3A	M	PHID
							MODELA	M	PHID
							MODELB	I	PHID
							MTX3A	O	PHID
							OUT	I	PHID
PHIO	ϕ_{old}	M	Bank angle from last nominal trajectory		/AEC03 /(	9)	AST3	M	PHIO
							MTX3A	I	PHIO
TIME	t	I	Time (elapsed)		/GENF /(	493)	ADICB3	O	TIME
							AST3	I	TIME
							BNTG	M	TIME
							CON3	I	TIME
							DTF3	I	TIME
							ENVPRM	I	TIME
							EQUA3	I	TIME
							FNTG	M	TIME
							MODELA	I	TIME
							OUT	I	TIME
							PDBC	I	TIME
							PROPIN	I	TIME
							REU3	M	TIME
							RKTA3A	M	TT
							RKTB3A	M	TT
							YREF3	M	TIME
TIMES	τ	I	Arc time	(SEC)	/GENF /(	319)	AST3	I	TIMES
							EQUA3	O	TIMES
							FNTG	M	TIMES
							GETIT	I	TIMES
							OUT	I	TIMES

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
VAR	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V
XL	$\lambda^{\Psi_i,\Omega_j}$	M	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTD3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
XLAMA	$\Lambda^{\Psi_i,\Omega_j}$	O	Impulse response function column vector associated with angle of attack	/AECO3 /(	16)	ADEQ3A	M	XLAMA
						ADIC3A	O	XLAMA
						AST3	O	XLAMA
						BGET3	O	XLAMA
						BSTD3	M	XLAMA
						MTX3A	I	XLAMA
						TRAN3	M	XLAMA
XLAMP	$\Lambda^{\Psi_i,\Omega_j}$	O	Impulse response function column vector associated with bank angle	/AECO3 /(	25)	ADEQ3A	M	XLAMP
						ADIC3A	O	XLAMP
						AST3	O	XLAMP
						BGET3	O	XLAMP
						BSTD3	M	XLAMP
						MTX3A	I	XLAMP
						TRAN3	M	XLAMP
YO	y_{old}	O	State vector of nominal trajectory	/STATE3/(	227)	AST3	O	YO
						MTX3A	I	YO

```
      SUBROUTINE AST3                                                   AST3
C                                                                       COMN
C         STORES  FORWARD TRAJECTORY DATA  FOR VARIOUS PURPOSES         COMN
C         ON  RANDOM ACCESS  AND  SEQUENTIAL   FILES                    COMN
C         ALSO  RETRIEVES DATA  FOR  FORWARD TRAJECTORY  (AGET)         COMN
C                                                                       COMN
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,IR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG~      ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED     ,                                           GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1Q      ,XK2Q      ,XK3Q     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,               MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)     -                          GENF
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2    ,                                STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE3D
     *UDV        ,VDG       ,GDS       ,RDG       ,PDG       ,ODG      , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                   .                           STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                     STATE3D
      COMMON/GLOBAL/                                                    GLOBAL
     *GR         ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM        GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC       ,NBRAN     ,NFARC    ,ID(4)      GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                              GLOBAL
     *,GM         ,PSIRF,IPFLG1,       IPFLG2,IPFLG3,IPFLG4,INEQFL(20)  GLOBAL
     *,ITPSO ,KSOL       ,KGLOBL(8)                                     RETAP
      COMMON/AEC03/                                                     AEC03
     *APHO       ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      , AEC03
     *SINA       ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI   , AEC03
     *COSPHI     ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDG      , AEC03
     *CDOM       ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLGM     , AEC03
     *CM         ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM      , AEC03
     *CLAM       ,CL        ,CLA       ,CLM       ,                      AEC03
     *CO         ,CDA       ,COM                                        AEC03
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                           RETREV
      COMMON/RETREV/                                                    RETREV
     *NBUFA(2)   ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA     , RETREV
     *MIXB       ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB      RETREV
     *,NBUFB     ,IBUFB                                                 RETREV
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP       ,ICOP      ,IFAW   .  ,IFAR      ,IFB       ,IND      , XCODES
```

```
      *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
      *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
      *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
      *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         , XCODES
      *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
      *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
      *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
      *NCTIN     ,NEOF ,ILAB(8),JPRP,JG(I,MTT,MPIN(20),JP1,JP2,JP3        XCODES
       DIMENSION D(990),E(990)                                            AST3
       DATA ZERO/0./                                                      JULY28
C        I     INITIALIZE  BUFFER  AND  INDEX COUNTERS                   COMN
       ENTRY  BEGWR                                                       AST3
       IBLK1 =0                                                           AST3
       IBUF1 =0                                                           AST3
       RETURN                                                             AST3
C                                                                         COMN
C     II  STORE  NO BUFFERS USED FOR  WHOLE  TRAJECTORY                   COMN
C           ALSO  NO  OF  WORDS  IN  LAST  BUFFER.                        COMN
C           IF  ANY  DATA IN LAST BUFFER,                                 COMN
C                DUMP  LAST  BUFFER  ONTO  STORAGE FILE                   COMN
       ENTRY ENDWR                                                        AST3
       MAXA(IFAW-38) =IBLK1                                               AST3
       NBUFA(IFAW-38)=IBUF1+1                                             AST3
       IF(IBLK1 NE.0) GO TO 20                                            AST3
       NBUFA(IFAW -38) = IBUF1                                            AST3
       MAXA(IFAW-38)= MXA                                                 AST3
       RETURN                                                             AST3
C                                                                         COMN
C      III     STORE  TRAJECTORY DATA IN  BUFFER                          COMN
       ENTRY  ASTO                                                        AST3
       D  (IBLK1+1)= IARC                                                 AST3
       D(IBLK1+2) =10000*IPST+100*JP1+JP3                                 AST3
       D  (IBLK1+3)= TIME                                                 AST3
       IBLK1=IBLK1+3                                                      AST3
       DO 10  I=1,NEQ                                                     AST3
       IBLK1 = IBLK1+1                                                    AST3
    10 D(IBLK1) = VAR(I)                                                  AST3
       IBLK1= IBLK1+1                                                     AST3
       D(IBLK1)  = APHR                                                   AST3
       IBLK1=IBLK1+1                                                      AST3
       D(IBLK1)= PHID                                                     AST3
C                                                                         COMN
C      III-A    IF  BUFFER FULL ,DUMP ON RANDOM FILE,ELSE RETURN          COMN
       IF(IBLK1.GE.MXA) GO TO 20                                          AST3
       RETURN                                                             AST3
C                                                                         COMN
C       III-B   INCREASE BUFFER COUNTER  AND DUMP BUFFER                  COMN
C               IF  COUNTER EXCEED MAX NO. OF  BUFFERS--ERROR RETURN      COMN
    20 IBUF1 =IBUF1 +1                                                    AST3
       IF(IBUF1  GT  NBFA-1) GO TO 30                                     SEP18
       CALL  WRITMS(IFAW,D,MIXA,IBUF1)                                    AST3
       IBLK1 =0                                                           AST3
       RETURN                                                             AST3
    30 ISTART = 6                                                         SEP18
       RETURN                                                             SEP18
C                                                                         COMN
C       IV     INTIALIZE  FORWARD READ OF RANDOM FILE OR                  COMN
C              SEQ FILE IF ON SOLUTION TRAJECTORY  (PRCO=1 OR 2)          COMN
       ENTRY  BGRA                                                        AST3
       IF(ITER EQ.3) REWIND  10                                           AST3
       IBUF2 =0                                                           AST3
       GO TO 90                                                           AST3
C                                                                         COMN
C       V     FETCH  TRAJECTORY  DATA                                     COMN
       ENTRY  AGETA                                                       AST3
C                                                                         COMN
C       V-A    STARTING NOMINAL  CONTROL ON FILE 11  GO TO VII            COMN
C              ELSE  GET  DATA FROM  BUFFER                               COMN
       IF(IND EQ.1) GO TO 100                                             RETAP
       ISTN = E(IBLK2 +1)                                                 AST3
       IPHN=E(IBLK2+2)/10000                                              AST3
C       V-B   NEED  CONTROL VECTOR  FLAGS  ON SOLUTION  TRAJECTORY        COMN
       IF(ITER NE.3) GO TO 40                                             AST3
       JP1=(E(IBLK2+2) -10000*IPHN)/10)                                   AST3
```

```
          JP3=E(IBLK2+2) - 10000*IPHN -100*JP1+.5                      AST3
       40 CONTINUE                                                     AST3
          FTIME= E(IBLK2 +3)                                           AST3
          IBLK2 = IBLK2 +3                                             AST3
          DO  50  I=1,NEQ                                              AST3
          IBLK2 = IBLK2+1                                              AST3
       50 YO(I) = E(IBLK2)                                             AST3
          IBLK2 = IBLK2+1                                              AST3
          APHO  = E(IBLK2)                                             AST3
          IBLK2 =IBLK2+1                                               AST3
          PHIO = E(IBLK2)                                              AST3
          IF(ITER.NE 3) GO TO 70                                       AST3
C          V-C     NEED TO RETRIEVE TRANSFORMED ADJOINTS ON            COMN
C                  SOLUTION  TRAJECTORY                                COMN
          IF(LUM EQ 0) GO TO 60                                        AST3
          READ(10) (XL(J,NCN),J=1,NEQ), XLAMA(NCN),XLAMP(NCN)          AST3
       60 CONTINUE                                                     AST3
C          V-D   IF ON SOLUTION TRAJ  CONTROL IS UNCHANGED             COMN
          APHR = APHO                                                  AST3
          PHIO= PHIO                                                   AST3
       70 CONTINUE                                                     AST3
          IF(IBUF2 EQ.NBUFA(IFAR-38))GO TO 80                          AST3
C                                                                      COMN
C         V-E  IF IN LAST BUFFER  TEST FOR LAST WORD IN PARTIAL BUFFER COMN
C              ELSE TEST FOR MAX NO OF WORDS IN BUFFER (MXA)           COMN
C              IF AT LAST WORD IN PARTIAL BUFFER  SET K=4 (INDICATES END)  COMN
          IF(IBLK2 GE.MXA) GO TO 90                                    AST3
          RETURN                                                       AST3
       80 IF(IBLK2 GE.MAXA(IFAR-38)) K=4                               AST3
          RETURN                                                       AST3
       90 IBUF2 = IBUF2 +1                                             AST3
C         V-F   READ IN  NEXT  BUFFER                                  COMN
          CALL  READMS(IFAR,E,MIXA,IBUF2)                              AST3
          IBLK2 =0                                                     AST3
          RETURN                                                       AST3
C                                                                      COMN
C         VI     WRITE  STARTING SOLUTION FOR QL OR NEXT CASE OF SD OR SIZING COMN
          ENTRY ASTOF                                                  AST3
          WRITE(11) IARC,TIME,TIMES,APHR,PHID,(VAR(I),I=1,NEQ), (XL(JJ,NCN), AST3
         *  JJ=1,NEQ),ZERO                                             JULY28
          RETURN                                                       AST3
C                                                                      COMN
C         VII   READ  STARTING NOMINAL CONTROL FROM FILE 11            COMN
      100 READ(11) ISTN,FTIME,DUM,APHR,PHID                            RETAP
          IPHN = IPH                                                   RETAP
          IF(EOF,11) 110,120                                           NOS
      110 K=4                                                          NOS
      120 RETURN                                                       NOS
          END                                                          AST3
```

20 OCT 72 6 01-46

SUBRØUTINE
BERØCØ

Subroutine BERØCØ

Purpose

Computes total aerodynamic coefficients for all aerodynamic options. Also computes partials of coefficients with respect to state and control.

Description

BERØCØ is called from subroutine VT.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALPHA	α	I	Angle of attack	(RAD)	/AEC03 /(	3)	BEROCO	I	ALPHA
							BLGCON	M	ALPHA
							BL2	I	ALPHA
							FNTG	O	ALPHA
							MAMECO	I	ALPHA
							MODELA	M	ALPHA
							MODELB	O	ALPHA
							REU3	O	ALPHA
							VT	I	ALPHA
APHR	α	I	Angle of attack	(DEG)	/AEC03 /(	2)	AGETB3	O	APHR
							AST3	M	APHR
							BEROCO	I	APHR
							BLGCON	O	APHR
							GUI3A	M	APHR
							MODELA	M	APHR
							MODELB	I	APHR
							MTX3A	O	APHR
							OUT	I	APHR
CD	C_D	O	Drag coefficient		/AEC03 /(	52)	BEROCO	O	CD
							OUT	I	CD
							VT	I	CD
CDA	$\partial C_D/\partial\alpha$	M	See symbol		/AEC03 /(	53)	BEROCO	M	CDA
							VT	I	CDA
CDM	$\partial C_D/\partial M$	O	See symbol		/AEC03 /(	54)	BEROCO	O	CDM
							VT	I	CDM
CDO	C_{D_0}	I	Drag coefficient at $\alpha = 0$		/AEC03 /(	34)	BEROCO	I	CDO
							EQUA3	I	CDO
CDOM	$\partial C_{D_0}/\partial M$	I	See symbol		/AEC03 /(	35)	BEROCO	I	CDOM
							EQUA3	I	CDOM
CL	C_L	M	Lift coefficient		/AEC03 /(	49)	BEROCO	M	CL
							OUT	I	CL
							VT	I	CL
CLA	C_{L_α}	M	Lift coefficient slope		/AEC03 /(	50)	BEROCO	M	CLA
							EQUA3	M	CLA
							VT	I	CLA
CLAM	$\partial C_{L_\alpha}/\partial M$	I	See symbol		/AEC03 /(	48)	BEROCO	I	CLAM
							EQUA3	M	CLAM
CLM	$\partial C_L/\partial M$	M	See symbol		/AEC03 /(	51)	BEROCO	M	CLM
							VT	I	CLM
CLO	C_{L_0}	I	Lift coefficient at $\alpha = 0$		/AEC03 /(	36)	BEROCO	I	CLO
							EQUA3	I	CLO
CLOM	$\partial C_{L_0}/\partial M$	I	See symbol		/AEC03 /(	40)	BEROCO	I	CLOM
							EQUA3	I	CLOM
DEG		I	Radian to angle conversion, 57.29577951		/DATA /(	2)	BEROCO	I	DEG
							BLGCON	I	RAD
							ENVPRM	I	RAD
							EQUA3	I	RAD
							FNTG	I	RAD
							GUI3A	I	RAD
							MODELA	I	RAD
							MTX3A	I	RAD
							OUT	I	RAD
							PADS1	O	RAD
							SDINP	I	RAD
							TRTOSZ	I	RAD
FK	k	I	Induced drag coefficient		/AEC03 /(	37)	BEROCO	I	FK
							EQUA3	I	FK
FKM	$\partial k/\partial M$	I	See symbol		/AEC03 /(	47)	BEROCO	I	FKM
							EQUA3	I	FKM

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IFOB		I	Forward or adjoint integration flag = 1 means forward = 2 means adjoint	/XCODES/(	178)	ACCEL	I	IFOB
						BEROCO	I	IFOB
						BLYNE	I	IFOB
						EQUA3	I	IFOB
						IMPUL	I	IFOB
						SPLYNE	I	IFOB
						TOPM	O	IFOB
JAER		I	Aerodynamic model option flag	/ARCDAT/(	9)	BEROCO	I	JAER
						EQUA3	I	JAER
						GEINP	I	JAER
						OUT	I	JAER
						PROPB	I	JAER
						PROPIN	I	JAER
						VT	I	JAER
MACH	m	I	Mach number	/GENF /(	307)	BEROCO	I	MACH
						ENVPRM	I	MACH
						EQUA3	M	MACH
						OUT	I	MACH

```
      SUBROUTINE BEROCO                                                     BEROCO
C                                                                           COMN
C        CALCULATES  AERO  COEFFICIENTS                                     COMN
C                                                                           COMN
      COMMON/ARCDAT/                                                        ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTP1      ,    ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      ,    ARCDAT
     *XLMAX     ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      ,    ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      ,    ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      ,    ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        ,    ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                     ARCDAT
     *  ,FRATE  ,  ARCD(9)                                                  RETAP
      DIMENSION  ARCDA(40)                                                  ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                               ARCDAT
      COMMON/GENF/                                                          GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   ,    GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       ,    GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS        ,    GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS        ,    GENF
     *VNO       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     ,    GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     ,    GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         ,    GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       ,    GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,    GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     ,    GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                          GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     ,    GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      ,    GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA     ,    GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,    GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       ,    GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                        GENF
      COMMON / GENF  /                                                      GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       ,    GENF
     *FRATED    ,IRATED                                                     GENF
     *P1        ,P2        ,P3       ,XK1        ,XK2       ,XK3       ,    GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,    GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,    GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,    GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      ,    GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,    GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPOY(3,8)       GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                      FRAT
      DIMENSION TPH1(10),TST1(10)                                           GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                    GENF
      COMMON /XCODES/                                                       XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2),   XCODES
     *JST   (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB     ,    XCODES
     *I20P      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       ,    XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    ,    XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS        ,    XCODES
     *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE     ,    XCODES
     *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         ,    XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     ,    XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         ,    XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB      ,    XCODES
     *NCTIN     ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3               XCODES
      COMMON/AEC03/                                                         AEC03
     *APHO      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       ,    AEC03
     *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI    ,    AEC03
     *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       ,    AEC03
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      ,    AEC03
     *CM        ,CMA       ,CMPM ,CMM ,CMO       ,CMOM      ,FKM       ,    AEC03
     *CLAM      ,CL        ,CLA       ,CLM       ,                          AEC03
     *CD        ,CDA       ,CDM                                             AEC03
      COMMON/DATA/                                                          DATA
     *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      ,    DATA
     *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4           DATA
      EQUIVALENCE(RAD,DEG)                                                  BEROCO
      IF(JAER .EQ. 2) GO TO 50                                              BEROCO   50
C      I      BIVARIATE DATA  TEST                                          COMN
      CL = CLO + ALPHA*CLA                                                  BEROCO
      CD = CDO + FK*CL**2                                                   BEROCO
```

```
      CDA = 2.*FK*CL*CLA                                    BEROCO
C                                                           COMN
C      II   ON FORWARD TRAJ. SKIP  PARTIALS                 COMN
      IF(IFOB.EQ.1) RETURN                                  BEROCO
      CLM = CLOM + ALPHA*CLAM                               BEROCO
      CDM = CDOM + CL*(FKM*CL + 2.*FK*CLM)                  BEROCO
      RETURN                                                BEROCO
C                                                           COMN
C      III  BIVARIATE AERO INTERPOLATION                    COMN
   50 CALL BLYNE(APHR ,MACH,CL)                             BEROCO
      CLA = CLA*DEG                                         BEROCO
      CDA = CDA*DEG                                         BEROCO
      RETURN                                                BEROCO
      END                                                   BEROCO
```

SUBRØUTINE BGET3

Subroutine BGET3

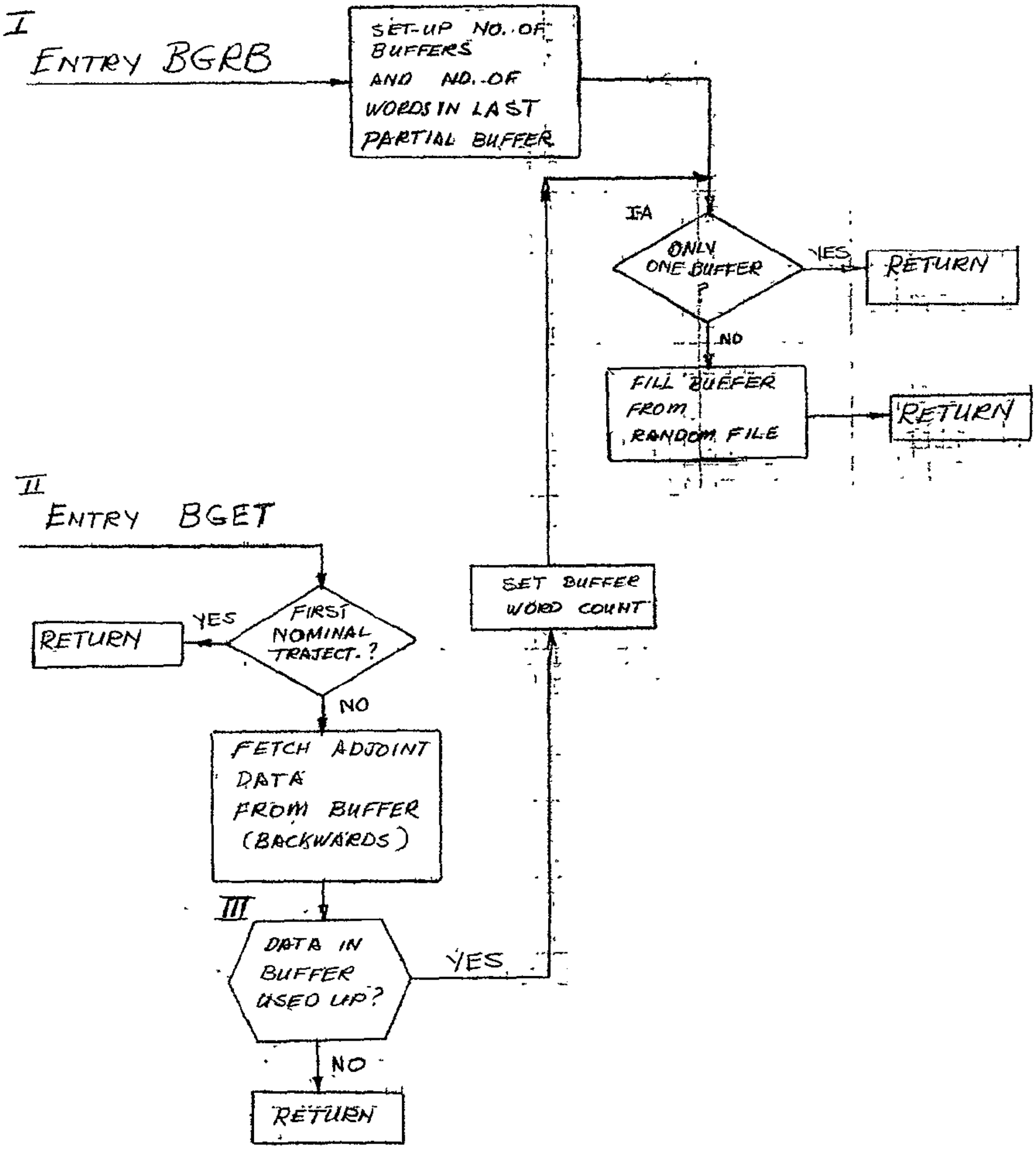
Purpose

This subroutine retrieves all adjoint data during forward trajectory. It has an initialization entry point called BGRB and a main entry point, BGET.

Description

Entries BGRB and BGET are called from various routines in the module.
BGET and BGRB are called from FNTG.
BGET is called from GETIT.
BGRB and BGET are called from TRAN3.

SUBROUTINE BGET3



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	A	O	Control integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BST03	I	A
						MTX3A	I	A
						PAY02	I	A
						SDINP	I	A
						TRAN3	I	A
IBLKB		M	Index of adjoint data buffer where either last word was stored or retrieved.	/RETREV/(	18)	BGET3	M	IBLKB
IBUFB		M	Counts number of buffers of adjoint data that have either been stored or retrieved as solution progress.	/RETREV/(	20)	BGET3	M	IBUFB
						BST03	M	IBUFB
IFB		I	File where adjoint solution is stored	/XCODES/(	140)	BGET3	I	IFB
						BST03	I	IFB
						TOPM	O	IFB
IND		I	Flag indicates whether on first nominal trajectory (IND=1)	/XCODES/(	141)	AST3	I	IND
						BGET3	I	IND
						FNTG	M	IND
						GUI3A	I	IND
						MTX3A	I	IND
						PROPIN	I	IND
MAXB		I	Number of words in last stored partial buffer of adjoint data. Corresponds to random file 41.	/RETREV/(	5)	BGET3	I	MAXB
						BST03	M	MAXB
MIXB		I	Maximum number of words in adjoint data buffer = 3000.	/RETREV/(	13)	BGET3	I	MIXB
						BST03	I	MIXB
						SDINP	I	MIXB
						TOPM	O	MIXB
MXB		I	Index of last stored word in full buffer of adjoint data.	/RETREV/(	15)	BGET3	I	MXB
						BST03	I	MXB
						SDINP	O	MXB
NBUFB		I	Number of buffers of adjoint data stored on last adjoint solution.	/RETREV/(	19)	BGET3	I	NBUFB
						BST03	O	NBUFB
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BST03	I	NCNST
						CON3	I	NCNST
						PAY02	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BST03	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
NPTB		I	Number of words stored at each adjoint solution time point.	/RETREV/(	17)	BGET3	I	NPTB
						SDINP	M	NPTB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
XL	$\lambda^{\Psi_i \Omega_j}$	O	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
XLAMA	$\Lambda^{\Psi_i \Omega_j}$	O	Impulse response function column vector associated with angle of attack	/AECO3 /(	16)	ADEQ3A	M	XLAMA
						ADIC3A	O	XLAMA
						AST3	O	XLAMA
						BGET3	O	XLAMA
						BSTO3	M	XLAMA
						MTX3A	I	XLAMA
						TRAN3	M	XLAMA
XLAMP	$\Lambda^{\Psi_i \Omega_j}$	O	Impulse response function column vector associated with bank angle	/AECO3 /(	25)	ADEQ3A	M	XLAMP
						ADIC3A	O	XLAMP
						AST3	O	XLAMP
						BGET3	O	XLAMP
						BSTO3	M	XLAMP
						MTX3A	I	XLAMP
						TRAN3	M	XLAMP
Z		I	Adjoint storage buffer	/BGET3 /(*	)	BGET3	I	Z

```
      SUBROUTINE BGET3                                                BGET3
C                                                                     COMN
C      RETRIEVE ADJOINTS , IMPULSE RESPONSE FUNCTION AND A MATRIX     COMN
C      FOR  TRAJECTORY CONTROL AND PARAMETER CALCULATIONS             COMN
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                         RETREV
      COMMON/RETREV/                                                  RETREV
     *NBUFA(2)  ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA     , RETREV
     *MIXB      ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB      RETREV
     *,NBUFB    ,IBUFB                                                RETREV
      COMMON/STATE3/                                                  STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10)  , STATE3D
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBF(15) , STATE3D
     *SINPSI    ,COSPSI    ,SINRHO    ,COSRHO    ,DCORHO    ,DCORO2   , STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                 STATE3D
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,QDV      , STATE3D
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,QDG      , STATE3D
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,QDR      , STATE3D
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP       ,PDP       ,ODP       ,UDP       ,VDD       ,GDD      , STATE3D
     *PDD       ,UDD       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                             STATE3D
      COMMON/STATE3/                                                  STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                    STATE3D
      COMMON/AECO3/                                                   AECO3
     *APHO      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      , AECO3
     *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI   , AECO3
     *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO      , AECO3
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLGM     , AECO3
     *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,EKM      , AECO3
     *CLAM      ,CL        ,CLA       ,CLM       ,                       AECO3
     *CD        ,CDA       ,CDM                                       AECO3
      COMMON /XCODES/                                                 XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE    , XCODES
     *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN     ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3        XCODES
      COMMON/GENF/                                                    GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT     ,CT        ,CALPHA    ,COE       ,DELTAE    ,SID      , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF /                                                 GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED    ,                                          GENF
     *P1        ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,            MACH, MACHR,        GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                FRAT
      DIMENSION TPH1(10),TST1(10)                                     GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                              GENF
```

```
      DIMENSION Z(3000)                                                 BGET3
C                                                                       COMN
C        I   INITIALIZE, SET-UP NO.OF BUFFERS AND NO.OF WORDS IN        COMN
C        LAST  (PARTIAL) BUFFER                                         COMN
      ENTRY  BGRB                                                       BGET3
      KK=  NCNST+1                                                      BGET3
      IBLKB =MAXB +1                                                    BGET3
      IBUFB =NBUFB+1                                                    BGET3
      GO TO  20                                                         BGET3
   10 IBLKB = MXB +1                                                    BGET3
   20 IBUFB = IBUFB -1                                                  BGET3
C        I-A    IF  ONLY  ONE  BUFFER  RETURN  ELSE READ IN CORRECT ONE COMN
C                                                                       COMN
      IF(IBUFB.LT.1) RETURN                                             BGET3
      CALL READMS(IFB,Z,MIXB,IBUFB)                                     BGET3
      RETURN                                                            BGET3
C                                                                       COMN
C        II    FETCH ADJOINT DATA FROM BUFFER  (SKIP IF STARTING SOLUTION) COMN
      ENTRY  BGET                                                       BGET3
      IF(IND.EQ.1) RETURN                                               NDS
      II= IBLKB - NPTB -1                                               BGET3
      DO 30 I=1,KK                                                      BGET3
      DO 30 J=1,NEQ                                                     BGET3
      II=II+1                                                           BGET3
   30 XL(J,I) = Z(II)                                                   BGET3
      DO 40  I=1,KK                                                     BGET3
      II= II+1                                                          BGET3
   40 XLAMA(I)= Z(II)                                                   BGET3
      DO 50  I=1,KK                                                     BGET3
      II=II+1                                                           BGET3
   50 XLAMP(I)= Z(II)                                                   BGET3
      DO 60 JJ= 1,KK                                                    BGET3
      DO 60 KI=1,JJ                                                     BGET3
      II=II +1                                                          BGET3
   60 A(KI,JJ)= Z(II)                                                   BGET3
      IBLKB = IBLKB -NPTB                                               BGET3
C                                                                       COMN
C        III   IF  DATA IN BUFFER USED UP LOAD BUFFER,ELSE RETURN       COMN
      IF(IBLKB.LE.1)  GO TO  10                                         BGET3
      RETURN                                                            BGET3
      END                                                               BGET3
```

20 OCT 72 G 01-46

SUBRØUTINE BLGCØN

Subroutine BLGCØN

Purpose

Subroutine BLGCØN solves for the in-plane control vector, w, and also during the adjoint solution computes the explicit and complete partials of the control vector with respect to state and control. The method of solution for the in-plane control vector is described in Section 9.2 of Volume I. The detailed logic flow description of this subroutine may be found in subroutine ALGCØN which is the Quasi-linearization module version of BLGCØN.

Description

BLGCØN is called both during the forward trajectory by subroutine MODELA and during the adjoint solution by subroutine MODELB.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
ALPHA	α	M	Angle of attack (RAD)	/AEC03 /(	3)	BEROCO	I	ALPHA
						BLGCON	M	ALPHA
						BL2	I	ALPHA
						FNTG	O	ALPHA
						MAMECO	I	ALPHA
						MODELA	M	ALPHA
						MODELB	O	ALPHA
						REU3	O	ALPHA
						VT	I	ALPHA
APHR	α	O	Angle of attack (DEG)	/AEC03 /(	2)	AGETB3	O	APHR
						AST3	M	APHR
						BEROCO	I	APHR
						BLGCON	O	APHR
						GUI3A	M	APHR
						MODELA	M	APHR
						MODELB	I	APHR
						MTX3A	O	APHR
						OUT	I	APHR
DELTAE	δ_E	M	Engine gimbal deflection angle (RAD)	/GENF /(	554)	BLGCON	M	DELTAE
						ELI	I	DELTAE
						OUT	I	DELTAE
						REU3	O	DELTAE
						VT	I	DELTAE
DET		W	Determinant	/BLGCON/(*	)	BLGCON	W	DET
DP	ΔW	I	Corrections to in-plane control vector	/BLGCON/(*	)	BLGCON	I	DP
DPDY	$\partial w/\partial y$	I	Matrix of partials of in-plane control vector wrt state	/GENF /(	610)	ACCEL	I	DPDY
						BLGCON	I	DPDY
						OUT	I	DPDY
ISTART		O	Initialization and divergance flag	/XCODES/(	147)	AST3	O	ISTART
						BLGCON	O	ISTART
						BLYNE	O	ISTART
						FNTG	I	ISTART
						MODELA	O	ISTART
						PROPIN	O	ISTART
						REU3	I	ISTART
						TEST	M	ISTART
						TOPM	M	ISTART
P		M	First element in in-plane control vector. Corresponds to thrust	/GENF /(	569)	BLGCON	M	P
						BLGCON	M	P1
						OUT	I	P1
P1		M	First element in in-plane control vector. Corresponds to thrust	/GENF /(	569)	BLGCON	M	P
						BLGCON	M	P1
						OUT	I	P1
P2		M	Second element in in-plane control vector. Corresponds to deflection	/GENF /(	570)	BLGCON	M	P2
P3		M	Third element in in-plane control vector, corresponds to α	/GENF /(	571)	BLGCON	M	P3
RAD		I	Radian to angle conversion, 57.29577951	/DATA /(	2)	BEROCO	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADS1	O	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD

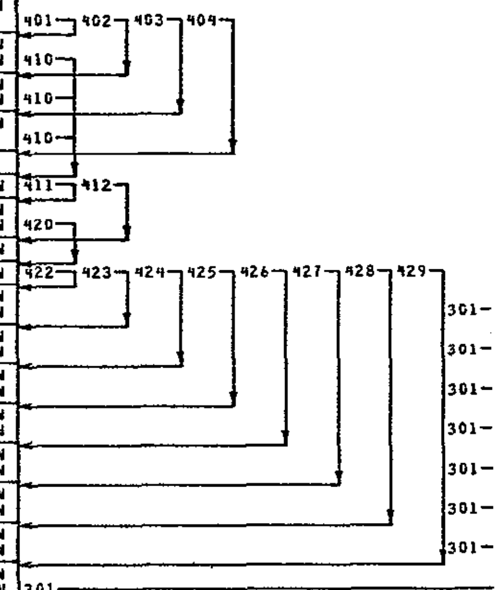
FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
	T	M	Thrust (LBS)	/GENF	/(411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SOER3	I	T
XK1		I	First control vector governing equation value corresponds to error in thrust	/GENF	/(572)	BLGCON	I	XK1
						FH1	O	XK1
						FH2	O	XK1
						FH3	O	XK1
						FH4	O	XK1
						MODELA	I	XK1
XK1A		I	Partial of governing equation wrt state or control vector component	/GENF	/(581)	BLGCON	I	XK1A
						FH3	M	XK1A
XK1D		I	Partial of governing equation wrt state or control vector component	/GENF	/(578)	BLGCON	I	XK1D
						FH3	M	XK1D
XK1T		I	Partial of governing equation wrt state or control vector component	/GENF	/(575)	BLGCON	I	XK1T
						FH1	O	XK1T
						FH2	O	XK1T
						FH3	M	XK1T
						FH4	O	XK1T
						MODELA	I	XK1T
XK1V		I	Partial of governing equation wrt state or control vector component	/GENF	/(584)	BLGCON	I	XK1V
						FH3	M	XK1V
						FH4	O	XK1V
XK2A		I	Partial of governing equation wrt state or control vector component	/GENF	/(582)	ACCEL	I	XK2A
						BLGCON	I	XK2A
						EL2	O	XK2A
XK2D		I	Partial of governing equation wrt state or control vector component	/GENF	/(579)	ACCEL	I	XK2D
						BLGCON	I	XK2D
						EL1	O	XK2D
						EL2	O	XK2D
XK2T		I	Partial of governing equation wrt state or control vector component	/GENF	/(576)	BLGCON	I	XK2T
						EL2	O	XK2T
XK3A		I	Partial of governing equation wrt state or control vector component	/GENF	/(583)	BLGCON	I	XK3A
						BL2	O	XK3A
						BL3	O	XK3A
						BL4	O	XK3A
						BL5	M	XK3A
						BL6	O	XK3A
						BL7	O	XK3A
						BL8	O	XK3A
XK3D		I	Partial of governing equation wrt state or control vector component	/GENF	/(580)	BLGCON	I	XK3D
						BL4	O	XK3D
						BL6	O	XK3D
						BL7	O	XK3D
						BL8	O	XK3D
XK3T		I	Partial of governing equation wrt state or control vector component	/GENF	/(577)	BLGCON	I	XK3T
						BL4	O	XK3T
						BL6	O	XK3T
						BL7	O	XK3T
						BL8	O	XK3T

```
      SUBROUTINE BLGCON(JP1,JP2,JP3)                                    P014
      COMMON/XCODES/IXCOD(146),ISTART                                   P014
      DIMENSION  DP(3), P(1)                                            BLGCON
      EQUIVALENCE (P,P1),(XKY,XK1V),(XKPI,XKPI11)                       BLGCON
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA     ,          DRAGV      ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED     ,                                           GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,             MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/AECO3/                                                     AECO3
     *APHO       ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      , AECO3
     *SINA       ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI   , AECO3
     *COSPHI     ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO      , AECO3
     *CDOM       ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM     , AECO3
     *CM         ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM      , AECO3
     *CLAM       ,CL        ,CLA       ,CLM       ,                      AECO3
     *CD         ,CDA       ,CDM                                        AECO3
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2    ,                                STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV      ,ODV       , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG      ,ODG       , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR      ,ODR       , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM      ,VDP       , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO      ,GDO       , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                            STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                    STATE3D
      DATA XALGCN /6HALGCON/, XKL11, XKL21, XKL12, XKL22, XKL13, XKL23  BLGCON
     */6*0 /                                                            BLGCON
      COMMON/DATA/                                                      DATA
     *PI         ,RAD       ,RDI       ,SC        ,UMF       ,TMPF     , DATA
     *FTNM       ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4       DATA
      EQUIVALENCE (XKPI21,XKPI11(2)),(XKPI31,XKPI11(3)),                P014
     * (XKPI12,XKPI11(4)),(XKPI22,XKPI11(5)),(XKPI32,XKPI11(6)),        P014
     * (XKPI13,XKPI11(7)),(XKPI23,XKPI11(8)),(XKPI33,XKPI11(9))         P014
      ASSIGN 302 TO JSWTCH                                              BLGCON
      DO 10 I=1,15                                                      P014
   10 P(I) = 0                                                          BLGCON
      P1 = T                                                            BLGCON
      P2 = DELTAE                                                       BLGCON
      P3 = ALPHA                                                        BLGCON
      ITR=0                                                             BLGCON
      IP3 = JP3 - 1                                                     BLGCON
      ASSIGN 100 TO ISWTCH                                              BLGCON
```

```
  100 CALL VTNOP                                                          BLGCON
      GO TO (101,102,103,104),JP1                                         FIXED
  101 CALL FH1001                                                         BLGCON
      GO TO 110                                                           BLGCON
  102 CALL FH2001                                                         BLGCON
      GO TO 110                                                           BLGCON
  103 CALL  FH3001                                                        BLGCON
      GO TO 110                                                           FIXED
  104 CALL  FH4001                                                        FIXED
  110 GO TO (111,112) ,JP2                                                BLGCON
  111 CALL EL1001                                                         BLGCON
      GO TO 120                                                           BLGCON
  112 CALL  EL2001                                                        BLGCON
  120 GO TO (122, 123, 124, 125, 126 , 127, 128 , 129 ),IP3               BLGCON
  122 CALL BL2001                                                         PO14
      GO TO 300                                                           BLGCON
  123 CALL BL3001                                                         BLGCON
      GO TO 300                                                           BLGCON
  124 CALL BL4001                                                         BLGCON
      GO TO 300                                                           BLGCON
  125 CALL BL5001                                                         BLGCON
      GO TO  300                                                          BLGCON
  126 CALL  BL6001                                                        BLGCON
      GO TO 300                                                           BLGCON
  127 CALL BL7001                                                         BLGCON
      GO TO 300                                                           BLGCON
  128 CALL BL8001                                                         BLGCON
      GO TO 300                                                           BLGCON
  129 CALL BL9001                                                         BLGCON
  300 ITR = ITR +1                                                        BLGCON
      IF(ITR GT 20) GO TO 3025                                            BLGCON
  301 CONTINUE                                                            BLGCON
      DET = XK2T*(XK1D*XK3A - XK1A*XK3D) + XK2D*(XK1A*XK3T - XK1T*XK3A)  BLGCON
     * + XK2A*(XK1T*XK3D - XK1D*XK3T)                                    BLGCON
      IF(ABS(DET) GT 1 E-20) GO TO 3011                                   BLGCON
 3025 ISTART=6                                                            BLGCON
      RETURN                                                              BLGCON
 3011 CONTINUE                                                            BLGCON
      XKPI11 = (XK2D*XK3A - XK2A*XK3D)/DET                                BLGCON
      XKPI21 = (XK2A*XK3T - XK2T*XK3A)/DET                                BLGCON
      XKPI31 = (XK2T*XK3D - XK2D*XK3T)/DET                                BLGCON
      XKPI12 = (XK1A*XK3D - XK1D*XK3A)/DET                                BLGCON
      XKPI22 = (XK1T*XK3A - XK1A*XK3T)/DET                                BLGCON
      XKPI32 = (XK1D*XK3T - XK1T*XK3D)/DET                                BLGCON
      XKPI13 = (XK1D*XK2A - XK1A*XK2D)/DET                                BLGCON
      XKPI23 = (XK1A*XK2T - XK1T*XK2A)/DET                                BLGCON
      XKPI33 = (XK1T*XK2D - XK1D*XK2T)/DET                                BLGCON
      GO TO JSWTCH                                                        BLGCON
  302 CALL MATMLT(DP, XKPI11, XK1, 3, 3, 1)                               BLGCON
      SUM = 0                                                             BLGCON
      DO 306 I = 1, 3                                                     BLGCON
      DIV = P(I)                                                          BLGCON
      IF(ABS(DIV) LT  01) DIV=.01                                         PO14
  306 SUM = SUM + ABS(DP(I)/DIV)                                          BLGCON
      IF(SUM  LE  1.E-12) GO TO 307                                       BLGCON
      CALL MATADD(P, P, DP, 3, 1)                                         BLGCON
      T = P1                                                              BLGCON
      DELTAE = P2                                                         BLGCON
      ALPHA = P3                                                          BLGCON
      APHR=ALPHA*RAD                                                      PO14
      GO TO ISWTCH                                                        BLGCON
  307 CALL MATADD(P, P, DP, 3, 1)                                         BLGCON
      T = P1                                                              BLGCON
      DELTAE = P2                                                         BLGCON
      ALPHA = P3                                                          BLGCON
```

20 OCT 72 G 01-46

```
       APHR=ALPHA*RAD                                              P014
       RETURN                                                      BLGCON
       ENTRY BLGCNV                                                BLGCON
       DO 308 I=16,65                                              P014
   308 P(I)=0.                                                     P014
       ASSIGN 500 TO JSWTCH                                        BLGCON
       CALL  VT                                                    BLGCON
       GO TO (401,402,403,404),JP1                                 FIXED
   401 CALL FH1                                                    BLGCON
       GO TO 410                                                   BLGCON
   402 CALL FH2                                                    BLGCON
       GO TO 410                                                   BLGCON
   403 CALL FH3                                                    BLGCON
       GO TO 410                                                   FIXED
   404 CALL FH4                                                    FIXED
   410 GO TO (411,412),JP2                                         BLGCON
   411 CALL EL1                                                    BLGCON
       GO TO 420                                                   BLGCON
   412 CALL  EL2                                                   BLGCON
   420 GO TO (422,422,423,424,425,426,427,428,429),JP3             BLGCON
   422 CALL BL2                                                    BLGCON
       GO TO 301                                                   BLGCON
   423 CALL BL3                                                    BLGCON
       GO TO 301                                                   BLGCON
   424 CALL  BL4                                                   BLGCON
       GO TO 301                                                   BLGCON
   425 CALL  BL5                                                   BLGCON
       GO TO 301                                                   BLGCON
   426 CALL  BL6                                                   BLGCON
       GO TO 301                                                   BLGCON
   427 CALL BL7                                                    BLGCON
       GO TO 301                                                   BLGCON
   428 CALL BL8                                                    BLGCON
       GO TO 301                                                   BLGCON
   429 CALL BL9                                                    BLGCON
       GO TO 301                                                   BLGCON
   500 CALL MATMLT(DPDY,XKPI11,XK1V,3,3,7)                         P014
       RETURN                                                      BLGCON
       END                                                         BLGCON
```



20 OCT 72 6.01-46

SUBRØUTINE
BLYNE

Subroutine BLYNE

<u>Purpose</u>

This routine interpolates bivariate tables.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ALFA		I	A 31 word array containing the mesh x_0, x_1, , x_n	/BICUBE/(	204)	BLICO	I	F
						BLICO	M	MACH
						BLYNE	I	ALFA
AMAX	x_N	I	The largest value of the first independent variable of a bivariate table.	/BICUBE/(	2)	BLICO	O	MMAX
						BLYNE	I	AMAX
						INBVAD	I	AMAX
AMIN	x_0	I	The smallest value of the first independent variable of a bivariate table.	/BICUBE/(	1)	BLICO	M	MMIN
						BLYNE	I	AMIN
						INBVAD	I	AMIN
C		O	A 32 word array containing the spline coefficients for the two bivariate functions at rectangle IRECT.	/BICUBE/(	12)	BLYNE	O	C
						BLYNE	I	CLOO
						INBVAD	O	C
CLOO		I	A 32 word array containing the spline coefficients for the two bivariate functions at rectangle IRECT	/BICUBE/(	12)	BLYNE	O	C
						BLYNE	I	CLOO
						INBVAD	O	C
IF		M	Last file in the grid in which interpolation occurred.	/BICUBE/(	3)	BLYNE	M	IF
						INBVAD	M	IF
IFMAX	N	I	Total number of files in grid.	/BICUBE/(	4)	BLICO	M	IRMAX
						BLYNE	I	IFMAX
						INBVAD	I	IFMAX
IFOB		I	Forward or adjoint integration flag = 1 means forward = 2 means adjoint	/XCODES/(	178)	ACCEL	I	IFOB
						BEROCO	I	IFOB
						BLYNE	I	IFOB
						EQUA3	I	IFOB
						IMPUL	I	IFOB
						SPLYNE	I	IFOB
						TOPM	O	IFOB
IR		M	Last rank in the grid in which interpolation occurred.	/BICUBE/(	7)	BLICO	M	IF
						BLYNE	M	IR
						INBVAD	M	IR
IREC		M	Logical record on IUNIT that contains spline coefficients for rectangle IRECT.	/BICUBE/(	11)	BLICO	M	IREC
						BLYNE	M	IREC
						INBVAD	M	IREC
IRECT		M	Grid rectangle associated with IR and IF.	/BICUBE/(	10)	BLICO	M	IRECT
						BLYNE	M	IRECT
						INBVAD	M	IRECT
IRMAX		I	Total number of ranks in grid.	/BICUBE/(	8)	BLICO	M	IFMAX
						BLYNE	I	IRMAX
						INBVAD	I	IRMAX
ISTART		O	Initialization and divergance flag	/XCODES/(	147)	AST3	O	ISTART
						BLGCON	O	ISTART
						BLYNE	O	ISTART
						FNTG	I	ISTART
						MODELA	O	ISTART
						PROPIN	O	ISTART
						REU3	I	ISTART
						TEST	M	ISTART
						TOPM	M	ISTART
IUNIT		I	Logical unit number on which bicubic spline coefficients are stored for this table.	/BICUBE/(	9)	BLICO	M	IUNIT
						BLYNE	I	IUNIT
						INBVAD	I	IUNIT
MACH		I	A 31 word array containing the mesh $y_0, y_1, \ldots, y_n$	/BICUBE/(	235)	BLICO	M	ALFA
						BLYNE	I	MACH
MMAX	y_M	I	The largest value of the second independent variable of a bivariate table.	/BICUBE/(	6)	BLICO	O	AMAX
						BLYNE	I	MMAX
MMIN	y_0	I	The smallest value of the second independent variable of a bivariate table.	/BICUBE/(	5)	BLICO	O	AMIN
						BLYNE	I	MMIN
T		I	A 160 word array containing logical record IREC.	/BICUBE/(	44)	BLICO	I	T
						BLYNE	I	T
						INBVAD	I	T

```
      SUBROUTINE BLYNE(A, M, U)                                         BLYNE
      REAL M, K, MMIN, MMAX, MACH                                       BLYNE
      COMMON/XCODES/ JKQ(177),IFOB                                      BLYNE
      EQUIVALENCE(JKQ(147),ISTART)                                      SEP18
      COMMON /BICUBE/ AMIN, AMAX, IF, IFMAX, MMIN, MMAX, IR, IRMAX,     BLYNE
     1IUNIT, IRECT, IREC, C(32), T(160), KNOTS(1)                       BLYNE
      DIMENSION U( 6), ALFA(1), MACH(1)                                 BLYNE
      EQUIVALENCE (ALFA, KNOTS), (MACH, KNOTS(32))                      BLYNE
      EQUIVALENCE                                                       BLYNE
     1(CL00, C( 1)), (CL01, C( 5)), (CL02, C( 9)), (CL03, C(13)),       BLYNE
     2(CL10, C( 2)), (CL11, C( 6)), (CL12, C(10)), (CL13, C(14)),       BLYNE
     3(CL20, C( 3)), (CL21, C( 7)), (CL22, C(11)), (CL23, C(15)),       BLYNE
     4(CL30, C( 4)), (CL31, C( 8)), (CL32, C(12)), (CL33, C(16)),       BLYNE
     5(CD00, C(17)), (CD01, C(21)), (CD02, C(25)), (CD03, C(29)),       BLYNE
     6(CD10, C(18)), (CD11, C(22)), (CD12, C(26)), (CD13, C(30)),       BLYNE
     7(CD20, C(19)), (CD21, C(23)), (CD22, C(27)), (CD23, C(31)),       BLYNE
     8(CD30, C(20)), (CD31, C(24)), (CD32, C(28)), (CD33, C(32))        BLYNE
      IF(MMIN .LE  M .AND. M .LE. MMAX) GO TO 20                        BLYNE
      ISTART = 6                                                        SEP18
      RETURN                                                            SEP18
   10 CONTINUE                                                          BLYNE
   20 IF(AMIN .LE  A  AND. A .LE. AMAX) GO TO 30                        BLYNE
      IF(A .LT  AMIN) A = AMIN                                          BLYNE
      IF(AMAX .LT. A) A = AMAX                                          BLYNE
   30 IF(M - MACH(IR)) 40,80,50                                         BLYNE
   40 IR = IR - 1                                                       BLYNE
      GO TO 30                                                          BLYNE
   50 IF(M - MACH(IR + 1)) 80,60,70                                     BLYNE
   60 IF(IR  GE  IRMAX) GO TO 80                                        BLYNE
   70 IR = IR + 1                                                       BLYNE
      GO TO 30                                                          BLYNE
   80 IF(A - ALFA(IF)) 90,130,100                                       BLYNE
   90 IF = IF - 1                                                       BLYNE
      GO TO 80                                                          BLYNE
  100 IF(A - ALFA(IF + 1)) 130,110,120                                  BLYNE
  110 IF(IF  GE  IFMAX) GO TO 130                                       BLYNE
  120 IF = IF + 1                                                       BLYNE
      GO TO 80                                                          BLYNE
  130 H = A - ALFA(IF)                                                  BLYNE
      K = M - MACH(IR)                                                  BLYNE
      JRECT = IR + IRMAX*(IF - 1)                                       BLYNE
      IF(JRECT  EQ. IRECT) GO TO 160                                    BLYNE
      IRECT = JRECT                                                     BLYNE
      JREC = (IRECT - 1)/5 + 2                                          BLYNE
      IF(JREC .EQ. IREC) GO TO 140                                      BLYNE
      IREC = JREC                                                       BLYNE
      CALL READMS(IUNIT, T, 160, IREC)                                  BLYNE
  140 IB = 32*(IRECT - 5*IREC + 9)                                      BLYNE
      DO 150 I = 1, 32                                                  BLYNE
      J = I + IB                                                        BLYNE
  150 C(I) = T(J)                                                       BLYNE
  160 CONTINUE                                                          BLYNE
      CL0 = CL00 + K*(CL01 + K*(CL02 + K*CL03))                         BLYNE
      CL1 = CL10 + K*(CL11 + K*(CL12 + K*CL13))                         BLYNE
      CL2 = CL20 + K*(CL21 + K*(CL22 + K*CL23))                         BLYNE
      CL3 = CL30 + K*(CL31 + K*(CL32 + K*CL33))                         BLYNE
      U(1) = CL0 + H*(CL1 + H*(CL2 + H*CL3))                            BLYNE
      CD0 = CD00 + K*(CD01 + K*(CD02 + K*CD03))                         BLYNE
      CD1 = CD10 + K*(CD11 + K*(CD12 + K*CD13))                         BLYNE
      CD2 = CD20 + K*(CD21 + K*(CD22 + K*CD23))                         BLYNE
      CD3 = CD30 + K*(CD31 + K*(CD32 + K*CD33))                         BLYNE
      U(4) = CD0 + H*(CD1 + H*(CD2 + H*CD3))                            BLYNE
      GO TO (180,170),IFOB                                              PO14
  170 CLOPP = K*CL03                                                    BLYNE
      CLOPP = CLOPP + CLOPP + CLOPP + CL02                              BLYNE
      CLOP = CL01 + K*(CL02 + CLOPP)                                    BLYNE
      CL1PP = K*CL13                                                    BLYNE
      CL1PP = CL1PP + CL1PP + CL1PP + CL12                              BLYNE
```

```
      CL1P = CL11 + K*(CL12 + CL1PP)                                    BLYNE
      CL2PP = K*CL23                                                    BLYNE
      CL2PP = CL2PP + CL2PP + CL2PP + CL22                              BLYNE
      CL2P = CL21 + K*(CL22 + CL2PP)                                    BLYNE
      CL3PP = K*CL33                                                    BLYNE
      CL3PP = CL3PP + CL3PP + CL3PP + CL32                              BLYNE
      CL3P = CL31 + K*(CL32 + CL3PP)                                    BLYNE
      UXX = H*CL3                                                       BLYNE
      UXX = UXX + UXX + UXX + CL2                                       BLYNE
      U(2) = CL1 + H*(CL2 + UXX)                                        BLYNE
      U(3) = CL0P + H*(CL1P + H*(CL2P + H*CL3P))                        BLYNE
      CD0PP = K*CD03                                                    BLYNE
      CD0PP = CD0PP + CD0PP + CD0PP + CD02                              BLYNE
      CD0P = CD01 + K*(CD02 + CD0PP)                                    BLYNE
      CD1PP = K*CD13                                                    BLYNE
      CD1PP = CD1PP + CD1PP + CD1PP + CD12                              BLYNE
      CD1P = CD11 + K*(CD12 + CD1PP)                                    BLYNE
      CD2PP = K*CD23                                                    BLYNE
      CD2PP = CD2PP + CD2PP + CD2PP + CD22                              BLYNE
      CD2P = CD21 + K*(CD22 + CD2PP)                                    BLYNE
      CD3PP = K*CD33                                                    BLYNE
      CD3PP = CD3PP + CD3PP + CD3PP + CD32                              BLYNE
      CD3P = CD31 + K*(CD32 + CD3PP)                                    BLYNE
      VXX = H*CD3                                                       BLYNE
      VXX = VXX + VXX + VXX + CD2                                       BLYNE
      U(5) = CD1 + H*(CD2 + VXX)                                        BLYNE
      U(6) = CD0P + H*(CD1P + H*(CD2P + H*CD3P))                        BLYNE
      RETURN                                                            BLYNE
  180 UXX = H*CL3                                                       BLYNE
      UXX = UXX + UXX + UXX + CL2                                       BLYNE
      U(2) = CL1 + H*(CL2 + UXX)                                        BLYNE
      VXX = H*CD3                                                       P014
      VXX = VXX + VXX + VXX + CD2                                       P014
      U(5) = CD1 + H*(CD2 + VXX)                                        P014
  190 RETURN                                                            BLYNE
      END                                                               BLYNE
```

20 OCT 72 6 01-46

Subroutines BL2 Through BL9

Purpose

Subroutine BL2 through BL9 supply (on option) the governing equation for the angle of attack, α , element of the in-plane control vector. This is governing equation K_3 described in Sections 9. 2 and 10 of Volume I. The correspondence between subroutine and steering options is listed below:

Subroutine Name	Steering Control Option
BL2	Constant angle of attack
BL3	Constant lift
BL4	Vertical rise of pitchover, $\mathring{\gamma}^* \neq 0$
BL5	Unpowered total acceleration limit
BL6	Gravity turn
BL7	Dynamic pressure limit
BL8	Heating rate limit
BL9	Reynolds number limit

Subroutines BL2 through BL9 are simplified versions of AL2 through AL9 used in the quasi-linearization module of PADS.

Each routine has a hierarchy of entry points. The entry points for example in BL7 include BL7010, BL7001 and BL 7000. The meaning of the numbers is similar in all of the "BL" routines and is spelled out below. (Subroutine BL7 is used only as an example).

Entry BL7000. Computes value of K_3, the governing equation.

Entry BL7001. First computes partials of K_3, with respect to elements of the in-plane control vector and then computes K_3.

Entry BL7010. Computes explicit partials of K_3 with respect to state, then computes partials of K_3 with respect to the in-plane control vector and finally computes K_3.

SUBRØUTINE
BL2

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALPHA	α	I	Angle of attack (RAD)	/AECO3 /(	3)	BEROCO	I	ALPHA
						BLGCON	M	ALPHA
						BL2	I	ALPHA
						FNTG	O	ALPHA
						MAMECO	I	ALPHA
						MODELA	M	ALPHA
						MODELB	O	ALPHA
						REU3	O	ALPHA
						VT	I	ALPHA
CALPHA		I	Constant value of angle-of-attack (RAD)	/GENF /(	552)	BL2	I	CALPHA
						MODELA	M	CALPHA
						MODELB	M	CALPHA
XK3		O	Third control vector governing equation value. Corresponds to error in algebraic equation involving α.	/GENF /(	574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3
XK3A		O	Partial of governing equation wrt state or control vector component	/GENF /(	583)	BLGCON	I	XK3A
						BL2	O	XK3A
						BL3	O	XK3A
						BL4	O	XK3A
						BL5	M	XK3A
						BL6	O	XK3A
						BL7	O	XK3A
						BL8	O	XK3A

BL2

```
      SUBROUTINE BL2                                                    BL2
      COMMON/AECO3/                                                     AECO3
     *APHO      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       , AECO3
     *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI    , AECO3
     *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       , AECO3
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      , AECO3
     *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       , AECO3
     *CLAM      ,CL        ,CLA       ,CLM       ,                 *      AECO3
     *CD        ,CDA       ,CDM                                         AECO3
      COMMON/GENF/                                                      GENF
     *DMG(20)   ,DMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP       , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SIO       , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                      GENF
      COMMON / GENF  /                                                  GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED    ,IRATED    ,                                            GENF
     *P1        ,P2        ,P3       ,XK1        ,XK2       ,XK3       , GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,   GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
C                                                                       BL2
      ENTRY  BL2001                                                     BL2
   40 XK3A = 1.                                                         BL2
      ENTRY  BL2000                                                     BL2
   50 XK3 = ALPHA - CALPHA                                              BL2
C                                                                       BL2
      RETURN                                                            BL2
      END                                                               BL2
```

20 OCT 72 G 01-46

SUBRØUTINE BL3

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK		LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CULFT	.	I	Constant value of ULFT	(LBS)	/GENF	/(	550)	BL3	I	CULFT
								MODELA	O	CULFT
ULFT	L_U	I	Untrimmed aero.lift		/GENF	/(	540)	BL3	I	ULFT
								MODELA	I	ULFT
								VT	M	ULFT
ULFTA		I	Partial of ULFT wrt angle of attack		/GENF	/(	543)	BL3	I	ULFTA
								VT	M	ULFTA
ULFTR		I	Partial of ULFT wrt altitude		/GENF	/(	542)	BL3	I	ULFTR
								VT	M	ULFTR
ULFTV		I	Partial of ULFT wrt velocity		/GENF	/(	541)	BL3	I	ULFTV
								VT	M	ULFTV
XK3		O	Third control vector governing equation value. Corresponds to error in algebraic equation involving α.		/GENF	/(	574)	BL2	O	XK3
								BL3	O	XK3
								BL4	O	XK3
								BL5	O	XK3
								BL6	O	XK3
								BL7	O	XK3
								BL8	O	XK3
								MODELA	I	XK3
								OUT	I	XK3
XK3A		O	Partial of governing equation wrt state or control vector component		/GENF	/(	583)	BLGCON	I	XK3A
								BL2	O	XK3A
								BL3	O	XK3A
								BL4	O	XK3A
								BL5	M	XK3A
								BL6	O	XK3A
								BL7	O	XK3A
								BL8	O	XK3A
XK3R		O	Partial of governing equation wrt state or control vector component		/GENF	/(	595)	BL3	O	XK3R
								BL4	O	XK3R
								BL5	M	XK3R
								BL6	O	XK3R
								BL7	O	XK3R
								BL8	O	XK3R
XK3V		O	Partial of governing equation wrt state or control vector component		/GENF	/(	586)	BL3	O	XK3V
								BL4	O	XK3V
								BL5	M	XK3V
								BL6	O	XK3V
								BL7	O	XK3V
								BL8	O	XK3V

```
      SUBROUTINE  BL3                                                   BL3
      COMMON/AECO3/                                                     AECO3
     *APHO      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      , AECO3
     *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI   , AECO3
     *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO      , AECO3
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM     , AECO3
     *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM      , AECO3
     *CLAM      ,CL        ,CLA       ,CLM       ,                      AECO3
     *CD        ,CDA       ,CDM                                         AECO3
      COMMON/GENF/                                                      GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TDS       ,TR(9)    , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT     ,CT        ,CALPHA    ,COE       ,DELTAE    ,SID      , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED    ,                                            GENF
     *P1        ,P2        ,P3        ,XK1       ,XK2       ,XK3      , GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1Q      ,XK2Q      ,XK3Q     , GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      ENTRY BL3010                                                      BL3
      XK3V = ULFTV                                                      BL3
      XK3R = ULFTR                                                      BL3
      ENTRY BL3001                                                      BL3
   40 XK3A = ULFTA                                                      BL3
      ENTRY BL3000                                                      BL3
   50 XK3 = ULFT - CULFT                                                BL3
C                                                                       BL3
      RETURN                                                            BL3
      END                                                               BL3
```

20 OCT 72 G.01-46

SUBRØUTINE BL4

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CODAE	$\cos(\alpha-\delta_E)$	I	See symbol		/GENF /(	549)	ACCEL	I	CODAE
							BL4	I	CODAE
							BL6	I	CODAE
							BL7	I	CODAE
							BL8	I	CODAE
							FH3	I	CODAE
							SDER3	I	CODAE
							VT	O	CODAE
COSA	$\cos\alpha$	I	See symbol		/AECO3 /(	8)	ACCEL	I	COSA
							BL4	I	COSA
							BL6	I	COSA
							BL7	I	COSA
							BL8	I	COSA
							FH3	I	COSA
							OUT	I	COSA
							VT	M	COSA
COSGAM	$\cos(\gamma)$	I	See symbol		/STATE3/(	687)	ACCEL	I	COSGAM
							BL4	I	COSGAM
							BL8	I	COSGAM
							DER3A	I	COSGAM
							EQUA3	O	COSGAM
							MODELA	I	COSGAM
							MODELB	I	COSGAM
							OUT	I	COSGAM
							PDBC	I	COSGAM
							PDY3A	I	COSGAM
COSPHI	$\cos\phi$	I	See symbol		/AECO3 /(	13)	ACCEL	I	COSPHI
							BL4	I	COSPHI
							GUI3A	M	COSPHI
							MODELA	M	COSPHI
							MODELB	M	COSPHI
							OUT	I	COSPHI
COSPSI	$\cos(\psi)$	I	See symbol		/STATE3/(	705)	BL4	I	COSPSI
							BL7	I	COSPSI
							BL8	I	COSPSI
							DER3A	I	COSPSI
							EQUA3	O	COSPSI
							MODELA	I	COSPSI
							MODELB	I	COSPSI
							PDBC	I	COSPSI
							PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol		/STATE3/(	707)	BL4	I	COSRHO
							BL7	I	COSRHO
							BL8	I	COSRHO
							DER3A	I	COSRHO
							EQUA3	M	COSRHO
							MODELA	I	COSRHO
							MODELB	I	COSRHO
							OUT	I	COSRHO
							PDBC	I	COSRHO
							PDY3A	I	COSRHO
COS2RO	$\cos(2\rho)$	I	See symbol		/STATE3/(	756)	BL4	I	COS2RO
							BL7	M	COS2RO
							BL8	M	COS2RO
							MODELA	O	COS2RO
							MODELB	O	COS2RO
DB	D_b	I	Base drag	(LBS)	/GENF /(	537)	ACCEL	I	DB
							BL4	I	DB
							BL6	I	DB
							BL7	I	DB
							BL8	I	DB
							EQUA3	I	DB
							FH3	I	DB
							OUT	I	DB
							SDER3	I	DB
							VT	I	DB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DBR		I	Partial of base drag wrt altitude		/GENF /(	536)	ACCEL	I	DBR
							BL4	I	DBR
							BL6	I	DBR
							BL7	I	DBR
							BL8	I	DBR
							EQUA3	I	DBR
							FH3	I	DBR
							VT	I	DBR
G	g	I	Gravitational attraction	(FT/SEC**2)	/GENF /(	301)	BL4	I	G
							BL7	I	G
							BL8	I	G
							DER3A	I	G
							EQUA3	M	G
							MODELA	I	G
							MODELB	I	G
							PDY3A	I	G
							SDER3	I	G
							SDINP	M	G
GAMMAD		I	Pitch rate	(RAD)	/GENF /(	564)	BL4	I	GAMMAD
							PROPB	O	GAMMAD
							PROPIN	O	GAMMAD
LIFT	L	I	Aerodynamic lift	(LBS)	/GENF /(	496)	ACCEL	I	LIFT
							BL4	I	LIFT
							BL5	I	LIFT
							BL6	I	LIFT
							ENVPRM	I	LIFT
							FH3	I	LIFT
							OUT	I	LIFT
							PROPB	O	LIFT
							PROPIN	O	LIFT
							VT	O	LIFT
LIFTA		I	Partial of lift wrt angle-of-attack		/GENF /(	531)	ACCEL	I	LIFTA
							BL4	I	LIFTA
							BL5	I	LIFTA
							BL6	I	LIFTA
							FH3	I	LIFTA
							VT	O	LIFTA
LIFTM		I	Partial of LIFT wrt mass		/GENF /(	535)	ACCEL	I	LIFTM
							BL4	I	LIFTM
							BL5	I	LIFTM
							BL6	I	LIFTM
							FH3	I	LIFTM
							VT	O	LIFTM
LIFTR		I	Partial of lift wrt altitude		/GENF /(	530)	ACCEL	I	LIFTR
							BL4	I	LIFTR
							BL5	I	LIFTR
							BL6	I	LIFTR
							FH3	I	LIFTR
							VT	O	LIFTR
LIFTV		I	Partial of lift wrt velocity		/GENF /(	529)	ACCEL	I	LIFTV
							BL4	I	LIFTV
							BL5	I	LIFTV
							BL6	I	LIFTV
							FH3	I	LIFTV
							VT	O	LIFTV
M	m	I	Mass		/STATE3/(	4)	ACCEL	I	M
							BL4	I	M
							BL8	I	M
							EQUA3	I	M
							OUT	I	M
							SDER3	I	M
OMEGA	ω	I	Earth rotation rate	(RAD/SEC)	/STATE3/(	719)	BL4	I	OMEGA
							BL7	I	OMEGA
							TOPM	O	OMEGA
OMEGA2	ω^2	I	See symbol		/STATE3/(	720)	BL4	I	OMEGA2
							BL7	I	OMEGA2
							BL8	I	OMEGA2
							TOPM	O	OMEGA2

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
PG		M	Partial of ϕ wrt state	/GENF /(	606)	BL4	M	PG
PO		M	Partial of ϕ wrt state	/GENF /(	609)	BL4	M	PO
PP		M	Partial of ϕ wrt state	/GENF /(	607)	BL4	M	PP
PR		M	Partial of ϕ wrt state	/GENF /(	608)	BL4	M	PR
PV		M	Partial of ϕ wrt state	/GENF /(	605)	BL4	M	PV
R	R	I	Radial distance from earth center to vehicle (FT)	/GENF /(	305)	BL4	I	R
						BL7	I	R
						BL8	I	R
						DER3A	I	R
						EQUA3	M	R
						MODELA	I	R
						MODELB	I	R
						PDBC	I	R
						PDY3A	I	R
						TRTOSZ	I	R
SIDAE	$\sin(\alpha-\delta_E)$	I	See symbol	/GENF /(	557)	ACCEL	I	SIDAE
						BL4	I	SIDAE
						BL6	I	SIDAE
						BL7	I	SIDAE
						BL8	I	SIDAE
						FH3	I	SIDAE
						VT	O	SIDAE
SINA	$\sin\alpha$	I	See symbol	/AECO3 /(	7)	ACCEL	I	SINA
						BL4	I	SINA
						BL6	I	SINA
						BL7	I	SINA
						BL8	I	SINA
						FH3	I	SINA
						GUI3A	M	SINA
						OUT	I	SINA
						VT	M	SINA
SINGAM	$\sin(\gamma)$	I	See symbol	/STATE3/(	688)	BL4	I	SINGAM
						BL7	I	SINGAM
						BL8	I	SINGAM
						DER3A	I	SINGAM
						EQUA3	O	SINGAM
						MODELA	I	SINGAM
						MODELB	I	SINGAM
						PDBC	I	SINGAM
						PDY3A	I	SINGAM
						SDER3	I	SINGAM
SINPHI	$\sin\phi$	I	See symbol	/AECO3 /(	12)	ACCEL	I	SINPHI
						BL4	I	SINPHI
						MODELA	M	SINPHI
						MODELB	M	SINPHI
						OUT	I	SINPHI
SINPSI	$\sin(\psi)$	I	See symbol	/STATE3/(	704)	BL4	I	SINPSI
						BL7	I	SINPSI
						BL8	I	SINPSI
						DER3A	I	SINPSI
						EQUA3	O	SINPSI
						MODELA	I	SINPSI
						MODELB	I	SINPSI
						PDBC	I	SINPSI
						PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol	/STATE3/(	706)	BL4	I	SINRHO
						BL7	I	SINRHO
						BL8	I	SINRHO
						DER3A	I	SINRHO
						EQUA3	O	SINRHO
						MODELA	I	SINRHO
						MODELB	I	SINRHO
						OUT	I	SINRHO
						PDBC	I	SINRHO
						PDY3A	I	SINRHO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SIN2RO	$\sin(2\rho)$	I	See symbol	/STATE3/(	755)	BL4	I	SIN2RO
						BL7	M	SIN2RO
						BL8	M	SIN2RO
						MODELA	O	SIN2RO
						MODELB	O	SIN2RO
T	T	I	Thrust (LBS)	/GENF /(	411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SDER3	I	T
V	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	D	KWOW
						YREF3	M	V
XKG	k_{γ}	I	Algebraic equation used in vertical rise and pitchover	/GENF /(	565)	BL4	I	XKG
						MODELA	M	XKGAM
						MODELB	M	XKGAM
XKP	k_{ψ}	I	Algebraic equation used in vertical rise and pitchover	/GENF /(	566)	BL4	I	XKP
						MODELA	M	XKPS
						MODELA	I	XKPSI
						MODELB	M	XKPS
						MODELB	I	XKPSI
XK3		O	Third control vector governing equation value Corresponds to error in algebraic equation involving α.	/GENF /(	574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
XK3A		O	Partial of governing equation wrt state or control vector component	/GENF	/(583)	BLGCON	I	XK3A
						BL2	O	XK3A
						BL3	O	XK3A
						BL4	O	XK3A
						BL5	M	XK3A
						BL6	O	XK3A
						BL7	O	XK3A
						BL8	O	XK3A
XK3D		O	Partial of governing equation wrt state or control vector component	/GENF	/(580)	BLGCON	I	XK3D
						BL4	O	XK3D
						BL6	O	XK3D
						BL7	O	XK3D
						BL8	O	XK3D
XK3G		O	Partial of governing equation wrt state or control vector component	/GENF	/(589)	BL4	O	XK3G
						BL7	O	XK3G
						BL8	O	XK3G
XK3M		O	Partial of governing equation wrt state or control vector component	/GENF	/(604)	BL4	O	XK3M
						BL5	M	XK3M
						BL6	O	XK3M
						BL7	O	XK3M
						BL8	O	XK3M
XK3O		O	Partial of governing equation wrt state or control vector component	/GENF	/(598)	BL4	O	XK3O
						BL7	O	XK3O
						BL8	O	XK3O
XK3P		O	Partial of governing equation wrt state or control vector component	/GENF	/(592)	BL4	O	XK3P
						BL7	O	XK3P
						BL8	O	XK3P
XK3R		O	Partial of governing equation wrt state or control vector component	/GENF	/(595)	BL3	O	XK3R
						BL4	O	XK3R
						BL5	M	XK3R
						BL6	O	XK3R
						BL7	O	XK3R
						BL8	O	XK3R
XK3T		O	Partial of governing equation wrt state or control vector component	/GENF	/(577)	BLGCON	I	XK3T
						BL4	O	XK3T
						BL6	O	XK3T
						BL7	O	XK3T
						BL8	O	XK3T
XK3V		O	Partial of governing equation wrt state or control vector component	/GENF	/(586)	BL3	O	XK3V
						BL4	O	XK3V
						BL5	M	XK3V
						BL6	O	XK3V
						BL7	O	XK3V
						BL8	O	XK3V

```
      SUBROUTINE  BL4                                                   BL4
      COMMON/AECO3/                                                     AECO3
     *APHO      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       ,AECO3
     *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI    ,AECO3
     *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       ,AECO3
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLGM      ,AECO3
     *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       ,AECO3
     *CLAM      ,CL        ,CLA       ,CLM       ,                      AECO3
     *CD        ,CDA       ,CDM                                         AECO3
      COMMON/GENF/                                                      GENF
     *OMG(20)   ,OMGP(20,2),VARD(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   ,GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       ,GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS        ,GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS        ,GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     ,GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     ,GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         ,GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       ,GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     ,GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     ,GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      ,GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA     ,GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       ,GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       ,GENF
     *FRATED    ,IRATED    ,                                            GENF
     *P1        ,P2        ,P3       ,XK1        ,XK2       ,XK3       ,GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      ,GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)   ,STATE3D
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) ,STATE3D
     *SINPSI    ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    ,STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                 STATE3D
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       ,STATE3D
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       ,STATE3D
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       ,STATE3D
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       ,STATE3D
     *GDP       ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       ,STATE3D
     *PDO       ,UDO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                      STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,  EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2), EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD), EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)            EQUV3
      REAL M,MU,MD                                                      EQUV3
C                                                                       BL4
      ASF(X, Y) = (XKG*Y - XKP*X)/SQUARE                                BL4
      ENTRY BL4010                                                      BL4
      ASSIGN 20 TO IGO                                                  BL4
      ASSIGN 5 TO LABL                                                  BL4
      GO TO 4                                                           BL4
      ENTRY BL4001                                                      BL4
      ASSIGN 40 TO IGO                                                  BL4
      GO TO 5                                                           BL4
      ENTRY BL4000                                                      BL4
      ASSIGN 50 TO IGO                                                  BL4
      GO TO 5                                                           BL4
    4 SQUARE = XKG**2 + XKP**2                                          BL4
      SQSQ = SQUARE**2                                                  BL4
      YVR = LIFTR - DBR*SINA                                            BL4
```

20 OCT 72 G 01-46

```
      XKGV = 2.*(V/R*COSGAM + OMEGA*COSRHO*SINPSI) - GAMMAD               BL4
      XKGG = (G - V**2/R)*SINGAM - R*OMEGA2*COSRHO*(COSRHO*SINGAM         BL4
     *        - SINRHO*COSPSI*COSGAM)                                     BL4
      XKGP = OMEGA*COSRHO*(2 *V*COSPSI - R*OMEGA*SINRHO*SINPSI*SINGAM)    BL4
      XKGR = (2.*G *R- V**2)/R**2*COSGAM + OMEGA2*COSRHO*(COSRHO*COSGAM   BL4
     *        + SINRHO*COSPSI*SINGAM)                                     BL4
      XKGO = -OMEGA*(R*OMEGA*(SIN2RO*COSGAM - COS2RO*COSPSI*SINGAM) + 2.  BL4
     *        *V*SINRHO*SINPSI)                                           BL4
      XKPV = 2.*(COSGAM*SINRHO*(V/(R*COSRHO)*COSGAM*SINPSI + OMEGA)       BL4
     *        - OMEGA*COSRHO*COSPSI*SINGAM)                               BL4
      XKPG = -2.*V*(SINRHO*SINGAM*(V/(R*COSRHO)*COSGAM*SINPSI + OMEGA)    BL4
     *        + OMEGA*COSRHO*COSPSI*COSGAM)                               BL4
      XKPP = (V*COSGAM)**2/(R*COSRHO)*SINRHO*COSPSI + OMEGA*COSRHO        BL4
     *        *(R*OMEGA*SINRHO*COSPSI + 2.*V*SINPSI*SINGAM)               BL4
      XKPR = SINRHO*SINPSI*(OMEGA2*COSRHO - (V/R*COSGAM)**2/COSRHO)       BL4
      XKPO = V*COSGAM*(V/(R*COSRHO**2)*COSGAM*SINPSI + 2.*OMEGA*COSRHO)   BL4
     *        + OMEGA*(R*OMEGA*SINPSI*COS2RO + 2.*V*SINRHO*COSPSI*SINGAM) BL4
      PV = ASF(XKGV, XKPV)                                                BL4
      PG = ASF(XKGG, XKPG)                                                BL4
      PP = ASF(XKGP, XKPP)                                                BL4
      PR = ASF(XKGR, XKPR)                                                BL4
      PO = ASF(XKGO, XKPO)                                                BL4
      CPV = -SINPHI*PV                                                    BL4
      CPG = -SINPHI*PG                                                    BL4
      CPP = -SINPHI*PP                                                    BL4
      CPR = -SINPHI*PR                                                    BL4
      CPO = -SINPHI*PO                                                    BL4
      GO TO LABL                                                          BL4
    5 TCDAE = T*CODAE                                                     BL4
      TSDAE = T*SIDAE                                                     BL4
      DBCA = DB*COSA                                                      BL4
      DBSA = DB*SINA                                                      BL4
      YY = TSDAE + LIFT - DBSA                                            BL4
      ZZ = TCDAE + LIFTA - DBCA                                           BL4
      GO TO IGO                                                           BL4
   20 XK3V = LIFTV*COSPHI + YY*CPV + M*XKGV                               BL4
      XK3G = YY*CPG + M*XKGG                                              BL4
      XK3P = YY*CPP + M*XKGP                                              BL4
      XK3R = YYR*COSPHI + YY*CPR + R*XKGR                                 BL4
      XK3O = YY*CPO + M*XKGO                                              BL4
      XK3M = XKG + LIFTA*COSPHI                                           BL4
   40 XK3T = COSPHI*SIDAE                                                 BL4
      XK3D = -TCDAE*COSPHI                                                BL4
      XK3A = ZZ*COSPHI                                                    BL4
   50 XK3 = YY*COSPHI + M*XKG                                             BL4
C                                                                         BL4
      RETURN                                                              BL4
      END                                                                 BL4
```

20 OCT 72 G.01-46

SUBRØUTINE BL5

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DRAG	D	I	Aerodynamic drag (LBS)	/GENF /(	497)	ACCEL	I	DRAG
						BL5	I	DRAG
						BL7	I	DRAG
						BL8	I	DRAG
						ENVPRM	I	DRAG
						FH3	I	DRAG
						OUT	I	DRAG
						PROPB	O	DRAG
						PROPIN	O	DRAG
						SDER3	I	DRAG
						VT	M	DRAG
DRAGA		I	Partial of drag wrt angle of attack	/GENF /(	534)	ACCEL	I	DRAGA
						BL5	I	DRAGA
						BL7	I	DRAGA
						BL8	I	DRAGA
						FH3	I	DRAGA
						VT	M	DRAGA
DRAGR		I	Partial drag wrt altitude	/GENF /(	533)	ACCEL	I	DRAGR
						BL5	I	DRAGR
						BL7	I	DRAGR
						BL8	I	DRAGR
						FH3	I	DRAGR
						VT	M	DRAGR
DRAGV		I	Partial of drag wrt velocity	/GENF /(	532)	ACCEL	I	DRAGV
						BL5	I	DRAGV
						BL7	I	DRAGV
						BL8	I	DRAGV
						FH3	I	DRAGV
						VT	M	DRAGV
GMAX	G_{MAX}	I	Maximum total acceleration g load	/ARCDAT/(	12)	BL5	I	GMAX
						FH3	I	GMAX
						MODELA	I	GMAX
						PROPB	I	GMAX
						PROPIN	I	GMAX
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
LIFT	L	I	Aerodynamic lift (LBS)	/GENF /(	496)	ACCEL	I	LIFT
						BL4	I	LIFT
						BL5	I	LIFT
						BL6	I	LIFT
						ENVPRM	I	LIFT
						FH3	I	LIFT
						OUT	I	LIFT
						PROPB	O	LIFT
						PROPIN	O	LIFT
						VT	O	LIFT
LIFTA		I	Partial of lift wrt angle-of-attack	/GENF /(	531)	ACCEL	I	LIFTA
						BL4	I	LIFTA
						BL5	I	LIFTA
						BL6	I	LIFTA
						FH3	I	LIFTA
						VT	O	LIFTA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
LIFTM		I	Partial of LIFT wrt mass	/GENF	/(535)	ACCEL	I	LIFTM
						BL4	I	LIFTM
						BL5	I	LIFTM
						BL6	I	LIFTM
						FH3	I	LIFTM
						VT	O	LIFTM
LIFTR		I	Partial of lift wrt altitude	/GENF	/(530)	ACCEL	I	LIFTR
						BL4	I	LIFTR
						BL5	I	LIFTR
						BL6	I	LIFTR
						FH3	I	LIFTR
						VT	O	LIFTR
LIFTV		I	Partial of lift wrt velocity	/GENF	/(529)	ACCEL	I	LIFTV
						BL4	I	LIFTV
						BL5	I	LIFTV
						BL6	I	LIFTV
						FH3	I	LIFTV
						VT	O	LIFTV
W	W	I	Weight (LBS)	/GENF	/(412)	BL5	I	W
						ENVPRM	I	W
						EQUA3	M	W
						FH3	I	W
						OUT	I	W
						PDBC	I	W
						REU3	I	W
						TRTOSZ	I	W
XK3		O	Third control vector governing equation value. Corresponds to error in algebraic equation involving α.	/GENF	/(574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3
XK3A		M	Partial of governing equation wrt state or control vector component	/GENF	/(583)	BLGCON	I	XK3A
						BL2	O	XK3A
						BL3	O	XK3A
						BL4	O	XK3A
						BL5	M	XK3A
						BL6	O	XK3A
						BL7	O	XK3A
						BL8	O	XK3A
XK3M		M	Partial of governing equation wrt state or control vector component	/GENF	/(604)	BL4	O	XK3M
						BL5	M	XK3M
						BL6	O	XK3M
						BL7	O	XK3M
						BL8	O	XK3M
XK3R		M	Partial of governing equation wrt state or control vector component	/GENF	/(595)	BL3	O	XK3R
						BL4	O	XK3R
						BL5	M	XK3R
						BL6	O	XK3R
						BL7	O	XK3R
						BL8	O	XK3R
XK3V		M	Partial of governing equation wrt state or control vector component	/GENF	/(586)	BL3	O	XK3V
						BL4	O	XK3V
						BL5	M	XK3V
						BL6	O	XK3V
						BL7	O	XK3V
						BL8	O	XK3V

```
      SUBROUTINE  BL5                                                      BL5
C                                                                          BL5
      COMMON/GLOBAL/                                                       GLOBAL
     *GR          ,ER         ,OMGZ       ,XLAMRF     ,YMURF      ,LUM     GLOBAL
     *,JJOP(10) ,IFATAL    ,NARC        ,NBRAN      ,NFARC      ,ID(4)    GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG     ,MAXTAB                                 GLOBAL
     *,GM         ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)      GLOBAL
     *,ITPSO  ,KSOL      ,KGLOBL(8)                                        RETAP
      COMMON/ARCDAT/                                                       ARCDAT
     *SREF        ,EJ         ,XISP       ,TMULT      ,DTNC       ,DTPI   , ARCDAT
     *IATM        ,IMODE      ,JAER       ,JPRO       ,QMAX       ,GMAX   , ARCDAT
     *XLMAX       ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX      ,MAEA   , ARCDAT
     *MAEB        ,MAEC       ,MAED       ,MAEE       ,MAEF       ,MAEG   , ARCDAT
     *MT          ,MISP       ,MXCG       ,MZCG       ,MWDA       ,MWDB   , ARCDAT
     *MDB         ,XCGR       ,ZCGR       ,XE         ,ZE         ,XT     , ARCDAT
     *DREF        ,MCND       ,RHOB       ,QMULT      ,REMAX              ARCDAT
     *  ,FRATE   ,  ARCD(9)                                                RETAP
      DIMENSION  ARCDA(40)                                                 ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                              ARCDAT
      COMMON/AECO3/                                                        AECO3
     *APHO        ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA    , AECO3
     *SINA        ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI , AECO3
     *COSPHI      ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDD    , AECO3
     *CDOM        ,CLO        ,FK         ,XCGM       ,ZCGM       ,CLOM   , AECO3
     *CM          ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM       ,FKM    , AECO3
     *CLAM        ,CL         ,CLA        ,CLM        ,                   AECO3
     *CD          ,CDA        ,CDM                                         AECO3
      COMMON/GENF/                                                         GENF
     *OMG(20)     ,OMGP(20,2),VARQ(9)     ,TOL(9)     ,SVAR(10)   ,WDC(20) , GENF
     *A(9,9)      ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)    ,DTP    , GENF
     *DTS         ,DT         ,G          ,DPSQ       ,Q          ,QS     , GENF
     *R           ,RE         ,MACH       ,PA         ,RO         ,CS     , GENF
     *VNU         ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ  , GENF
     *SVSQ        ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)  , GENF
     *TST(20)     ,TPH    (20),DIS(20)    ,DIP(20)    ,T          ,W      , GENF
     *TLP1(20)    ,TLS1   (20),DIP1(20)   ,DIS1(20)   ,TIME       ,DMP    , GENF
     *TIMPR       ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20) , GENF
     *AE          ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV  , GENF
     *QR          ,QV         ,FVAC       ,LIFTV      ,                   GENF
     *LIFTR       ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA  , GENF
     *             LIFTM      ,DBR        ,DB         ,ISP        ,ISPF   , GENF
     *                         ULFT       ,ULFTV      ,ULFTR      ,ULFTA  , GENF
     *XMCG        ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE  , GENF
     *CULFT       ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SID    , GENF
     *COD         ,SIDAE      ,XCG        ,ZCG        ,XJ                 GENF
      COMMON / GENF  /                                                     GENF
     *XJV         ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP    , GENF
     *FRATED     ,IRATED      ,                                            GENF
     *P1          ,P2         ,P3        ,XK1         ,XK2        ,XK3    , GENF
     *XK1T        ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D   , GENF
     *XK1A        ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V   , GENF
     *XK1G        ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P   , GENF
     *XK1R        ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O   , GENF
     *XK1U        ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M   , GENF
     *PV          ,PG         ,PP         ,PR         ,PO         ,DPDY(3,8) GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV  , IRATED                                    FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      ENTRY BL5010                                                         BL5
      XK3V = LIFT*LIFTV + DRAG*DRAGV                                       BL5
      XK3V = XK3V + XK3V                                                   BL5
      XK3R = LIFT*LIFTR + DRAG*DRAGR                                       BL5
      XK3R = XK3R + XK3R                                                   BL5
      XK3M = LIFT*LIFTM - GR*GMAX**2*W                                     BL5
      XK3M = XK3M + XK3M                                                   BL5
      ENTRY BL5001                                                         BL5
   40 XK3A = LIFT*LIFTA + DRAG*DRAGA                                       BL5
      XK3A = XK3A + XK3A                                                   BL5
      ENTRY BL5000                                                         BL5
   50 TERM3 = GMAX*W                                                       BL5
      XK3 = LIFT**2 + DRAG**2 - TERM3**2                                   BL5
C                                                                          BL5
      RETURN                                                               BL5
      END                                                                  BL5
```

20 OCT 72 6.01-46

SUBRØUTINE
BL6

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CODAE	$\cos(\alpha-\delta_E)$	I	See symbol		/GENF	/(549)	ACCEL	I	CODAE
							BL4	I	CODAE
							BL6	I	CODAE
							BL7	I	CODAE
							BL8	I	CODAE
							FH3	I	CODAE
							SDER3	I	CODAE
							VT	O	CODAE
COSA	$\cos\alpha$	I	See symbol		/AECO3	/(8)	ACCEL	I	COSA
							BL4	I	COSA
							BL6	I	COSA
							BL7	I	COSA
							BL8	I	COSA
							FH3	I	COSA
							OUT	I	COSA
							VT	M	COSA
DB	D_b	I	Base drag	(LBS)	/GENF	/(537)	ACCEL	I	DB
							BL4	I	DB
							BL6	I	DB
							BL7	I	DB
							BL8	I	DB
							EQUA3	I	DB
							FH3	I	DB
							OUT	I	DB
							SDER3	I	DB
							VT	I	DB
DBR		I	Partial of base drag wrt altitude		/GENF	/(536)	ACCEL	I	DBR
							BL4	I	DBR
							BL6	I	DBR
							BL7	I	DBR
							BL8	I	DBR
							EQUA3	I	DBR
							FH3	I	DBR
							VT	I	DBR
LIFT	L	I	Aerodynamic lift	(LBS)	/GENF	/(496)	ACCEL	I	LIFT
							BL4	I	LIFT
							BL5	I	LIFT
							BL6	I	LIFT
							ENVPRM	I	LIFT
							FH3	I	LIFT
							OUT	I	LIFT
							PROPB	O	LIFT
							PROPIN	O	LIFT
							VT	O	LIFT
LIFTA		I	Partial of lift wrt angle-of-attack		/GENF	/(531)	ACCEL	I	LIFTA
							BL4	I	LIFTA
							BL5	I	LIFTA
							BL6	I	LIFTA
							FH3	I	LIFTA
							VT	O	LIFTA
LIFTM		I	Partial of LIFT wrt mass		/GENF	/(535)	ACCEL	I	LIFTM
							BL4	I	LIFTM
							BL5	I	LIFTM
							BL6	I	LIFTM
							FH3	I	LIFTM
							VT	O	LIFTM
LIFTR		I	Partial of lift wrt altitude		/GENF	/(530)	ACCEL	I	LIFTR
							BL4	I	LIFTR
							BL5	I	LIFTR
							BL6	I	LIFTR
							FH3	I	LIFTR
							VT	O	LIFTR
LIFTV		I	Partial of lift wrt velocity		/GENF	/(529)	ACCEL	I	LIFTV
							BL4	I	LIFTV
							BL5	I	LIFTV
							BL6	I	LIFTV
							FH3	I	LIFTV
							VT	O	LIFTV

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
SIDAE	$\sin(\alpha-\delta_E)$	I	See symbol	/GENF /(	557)	ACCEL	I	SIDAE
						BL4	I	SIDAE
						BL6	I	SIDAE
						BL7	I	SIDAE
						BL8	I	SIDAE
						FH3	I	SIDAE
						VT	O	SIDAE
SINA	$\sin\alpha$	I	See symbol	/AEC03 /(	7)	ACCEL	I	SINA
						BL4	I	SINA
						BL6	I	SINA
						BL7	I	SINA
						BL8	I	SINA
						FH3	I	SINA
						GUI3A	M	SINA
						OUT	I	SINA
						VT	M	SINA
T	T	I	Thrust (LBS)	/GENF /(	411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SDER3	I	T
XK3		O	Third control vector governing equation value. Corresponds to error in algebraic equation involving α.	/GENF /(	574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3
XK3A		O	Partial of governing equation wrt state or control vector component	/GENF /(	583)	BLGCON	I	XK3A
						BL2	O	XK3A
						BL3	O	XK3A
						BL4	O	XK3A
						BL5	M	XK3A
						BL6	O	XK3A
						BL7	O	XK3A
						BL8	O	XK3A
XK3D		O	Partial of governing equation wrt state or control vector component	/GENF /(	580)	BLGCON	I	XK3D
						BL4	O	XK3D
						BL6	O	XK3D
						BL7	O	XK3D
						BL8	O	XK3D
XK3M		O	Partial of governing equation wrt state or control vector component	/GENF /(	604)	BL4	O	XK3M
						BL5	M	XK3M
						BL6	O	XK3M
						BL7	O	XK3M
						BL8	O	XK3M
XK3R		O	Partial of governing equation wrt state or control vector component	/GENF /(	595)	BL3	O	XK3R
						BL4	O	XK3R
						BL5	M	XK3R
						BL6	O	XK3R
						BL7	O	XK3R
						BL8	O	XK3R

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
XK3T		O	Partial of governing equation wrt state or control vector component	/GENF	/(577)	BLGCON	I	XK3T
						BL4	O	XK3T
						BL6	O	XK3T
						BL7	O	XK3T
						BL8	O	XK3T
XK3V		O	Partial of governing equation wrt state or control vector component	/GENF	/(586)	BL3	O	XK3V
						BL4	O	XK3V
						BL5	M	XK3V
						BL6	O	XK3V
						BL7	O	XK3V
						BL8	O	XK3V

BL6

```
      SUBROUTINE  BL6                                                      BL6
      COMMON/GENF/                                                         GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)  ,TOL(9)   ,SVAR(10) ,WDC(20)   ,  GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)  ,COTI(9,9),DCON(9)  ,DTP       ,  GENF
     *DTS       ,DT        ,G        ,DPSQ     ,Q        ,QS        ,  GENF
     *R         ,RE        ,MACH     ,PA       ,RO       ,CS        ,  GENF
     *VNU       ,PAR       ,ROR      ,CSR      ,VNR      ,SUMSQ     ,  GENF
     *SVSQ      ,TIMEPH    ,TIMES    ,TOP      ,TOS      ,TR(9)     ,  GENF
     *TST(20)   ,TPH   (20),DIS(20)  ,DIP(20)  ,T        ,W         ,  GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20) ,DIS1(20) ,TIME     ,OMP       ,  GENF
     *TIMPR     ,LIFT      ,DRAG     ,TAX      ,TBURN    ,TBU(20)   ,  GENF
     *AE        ,FP        ,FPOLD    ,FPD      ,MACHR    ,MACHV     ,  GENF
     *QR        ,QV        ,FVAC     ,LIFTV    ,                       GENF
     *LIFTR     ,LIFTA     ,          DRAGV    ,DRAGR    ,DRAGA     ,  GENF
     *           LIFTM     ,DBR      ,DB       ,ISP      ,ISPF      ,  GENF
     *                      ULFT     ,ULFTV    ,ULFTR    ,ULFTA     ,  GENF
     *XMCG      ,XMCGV     ,XMCGR    ,XMCGA    ,XMCGM    ,CODAE     ,  GENF
     *CULFT     ,CT        ,CALPHA   ,CDE      ,DELTAE   ,SID       ,  GENF
     *COD       ,SIDAE     ,XCG      ,ZCG      ,XJ                     GENF
      COMMON / GENF  /                                                     GENF
     *XJV       ,XJR       ,GH       ,GAMMAD   ,XKG      ,XKP       ,  GENF
     *FRATED    ,IRATED    ,                                               GENF
     *P1        ,P2        ,P3      ,XK1       ,XK2      ,XK3       ,  GENF
     *XK1T      ,XK2T      ,XK3T     ,XK1D     ,XK2D     ,XK30      ,  GENF
     *XK1A      ,XK2A      ,XK3A     ,XK1V     ,XK2V     ,XK3V      ,  GENF
     *XK1G      ,XK2G      ,XK3G     ,XK1P     ,XK2P     ,XK3P      ,  GENF
     *XK1R      ,XK2R      ,XK3R     ,XK10     ,XK2D     ,XK30      ,  GENF
     *XK1U      ,XK2U      ,XK3U     ,XK1M     ,XK2M     ,XK3M      ,  GENF
     *PV        ,PG        ,PP       ,PR       ,PD       ,DPDY(3,8)    GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,              MACH, MACHR,        GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                     FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      COMMON/AEC03/                                                        AEC03
     *APHO      ,APHR      ,ALPHA    ,VDA      ,GDA      ,PDA       ,  AEC03
     *SINA      ,COSA      ,PHIO     ,PHID     ,PHI      ,SINPHI    ,  AEC03
     *COSPHI    ,GDPH      ,PDPH     ,XLAMA(9) ,XLAMP(9) ,CDO       ,  AEC03
     *CDOM      ,CLO       ,FK       ,XCGM     ,ZCGM     ,CLOM      ,  AEC03
     *CM        ,CMA       ,CMAM ,CMM ,CMO     ,CMOM     ,FKM       ,  AEC03
     *CLAM      ,CL        ,CLA      ,CLM      ,                       AEC03
     *CD        ,CDA       ,CDM                                            AEC03
      ENTRY BL6010                                                         BL6
      ASSIGN 20 TO IGO                                                     BL6
      GO TO 5                                                              BL6
      ENTRY BL6001                                                         BL6
      ASSIGN 40 TO IGO                                                     BL6
      GO TO 5                                                              BL6
      ENTRY BL6000                                                         BL6
      ASSIGN 50 TO IGO                                                     BL6
    5 TCDAE = T*CODAE                                                      BL6
      TSDAE = T*SIDAE                                                      BL6
      DBCA = DB*COSA                                                       BL6
      DBSA = DB*SINA                                                       BL6
      GO TO IGO                                                            BL6
   20 XK3V = LIFTV                                                         BL6
      XK3R = LIFTR - DBR*SINA                                              BL6
      XK3M = LIFTM                                                         BL6
   40 XK3T = SIDAE                                                         BL6
      XK3D = -TCDAE                                                        BL6
      XK3A = TCDAE - DBCA + LIFTA                                          BL6
   50 XK3 = TSDAE + LIFT - DBSA                                            BL6
C                                                                          BL6
      RETURN                                                               BL6
      END                                                                  BL6
```

20 OCT 72 6.01-46

SUBRØUTINE BL7

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CODAE	$\cos(\alpha-\delta_E)$	I	See symbol		/GENF /(	549)	ACCEL	I	CODAE
							BL4	I	CODAE
							BL6	I	CODAE
							BL7	I	CODAE
							BL8	I	CODAE
							FH3	I	CODAE
							SDER3	I	CODAE
							VT	O	CODAE
COSA	$\cos\alpha$	I	See symbol		/AECO3 /(	8)	ACCEL	I	COSA
							BL4	I	COSA
							BL6	I	COSA
							BL7	I	COSA
							BL8	I	COSA
							FH3	I	COSA
							OUT	I	COSA
							VT	M	COSA
COSPSI	$\cos(\psi)$	I	See symbol		/STATE3/(	705)	BL4	I	COSPSI
							BL7	I	COSPSI
							BL8	I	COSPSI
							DER3A	I	COSPSI
							EQUA3	O	COSPSI
							MODELA	I	COSPSI
							MODELB	I	COSPSI
							PDBC	I	COSPSI
							PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol		/STATE3/(	707)	BL4	I	COSRHO
							BL7	I	COSRHO
							BL8	I	COSRHO
							DER3A	I	COSRHO
							EQUA3	M	COSRHO
							MODELA	I	COSRHO
							MODELB	I	COSRHO
							OUT	I	COSRHO
							PDBC	I	COSRHO
							PDY3A	I	COSRHO
COS2RO	$\cos(2\rho)$	M	See symbol		/STATE3/(	756)	BL4	I	COS2RO
							BL7	M	COS2RO
							BL8	M	COS2RO
							MODELA	O	COS2RO
							MODELB	O	COS2RO
DB	D_b	I	Base drag	(LBS)	/GENF /(	537)	ACCEL	I	DB
							BL4	I	DB
							BL6	I	DB
							BL7	I	DB
							BL8	I	DB
							EQUA3	I	DB
							FH3	I	DB
							OUT	I	DB
							SDER3	I	DB
							VT	I	DB
DBR		I	Partial of base drag wrt altitude		/GENF /(	536)	ACCEL	I	DBR
							BL4	I	DBR
							BL6	I	DBR
							BL7	I	DBR
							BL8	I	DBR
							EQUA3	I	DBR
							FH3	I	DBR
							VT	I	DBR
DRAG	D	I	Aerodynamic drag	(LBS)	/GENF /(	497)	ACCEL	I	DRAG
							BL5	I	DRAG
							BL7	I	DRAG
							BL8	I	DRAG
							ENVPRM	I	DRAG
							FH3	I	DRAG
							OUT	I	DRAG
							PROPB	O	DRAG
							PROPIN	O	DRAG
							SDER3	I	DRAG
							VT	M	DRAG

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DRAGA		I	Partial of drag wrt angle of attack	/GENF /(	534)	ACCEL	I	DRAGA
						BL5	I	DRAGA
						BL7	I	DRAGA
						BL8	I	DRAGA
						FH3	I	DRAGA
						VT	M	DRAGA
DRAGR		I	Partial drag wrt altitude	/GENF /(	533)	ACCEL	I	DRAGR
						BL5	I	DRAGR
						BL7	I	DRAGR
						BL8	I	DRAGR
						FH3	I	DRAGR
						VT	M	DRAGR
DRAGV		I	Partial of drag wrt velocity	/GENF /(	532)	ACCEL	I	DRAGV
						BL5	I	DRAGV
						BL7	I	DRAGV
						BL8	I	DRAGV
						FH3	I	DRAGV
						VT	M	DRAGV
G	g	I	Gravitational attraction (FT/SEC**2)	/GENF /(	301)	BL4	I	G
						BL7	I	G
						BL8	I	G
						DER3A	I	G
						EQUA3	M	G
						MODELA	I	G
						MODELB	I	G
						PDY3A	I	G
						SDER3	I	G
						SDINP	M	G
GH		I	Partial of gravity wrt altitude	/GENF /(	563)	BL7	I	GH
						BL8	I	GH
						EQUA3	O	GH
						PDY3A	I	DGDH
OMEGA	ω	I	Earth rotation rate (RAD/SEC)	/STATE3/(	719)	BL4	I	OMEGA
						BL7	I	OMEGA
						TOPM	O	OMEGA
OMEGA2	ω^2	I	See symbol	/STATE3/(	720)	BL4	I	OMEGA2
						BL7	I	OMEGA2
						BL8	I	OMEGA2
						TOPM	O	OMEGA2
R	R	I	Radial distance from earth center to vehicle (FT)	/GENF /(	305)	BL4	I	R
						BL7	I	R
						BL8	I	R
						DER3A	I	R
						EQUA3	M	R
						MODELA	I	R
						MODELB	I	R
						PDBC	I	R
						PDY3A	I	R
						TRTOSZ	I	R
RO	ρ_a	I	Atmospheric density (SLUGS/FT**3)	/GENF /(	309)	BL7	I	RO
						BL8	I	RO
						DER3A	I	RO
						EQUA3	I	RO
						OUT	I	RO
						PDBC	I	RO
						PDY3A	I	RO
ROR		I	Deriv Of density wrt alt.	/GENF /(	313)	BL7	I	ROR
						BL8	I	ROR
						EQUA3	I	ROR
						PDBC	I	ROR
						PDY3A	I	ROR
SIDAE	$\sin(\alpha-\delta_E)$	I	See symbol	/GENF /(	557)	ACCEL	I	SIDAE
						BL4	I	SIDAE
						BL6	I	SIDAE
						BL7	I	SIDAE
						BL8	I	SIDAE
						FH3	I	SIDAE
						VT	O	SIDAE

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBR	CODE	VAR
SINA	$\sin\alpha$	I	See symbol		/AEC03 /(	7)	ACCEL	I	SINA
							BL4	I	SINA
							BL6	I	SINA
							BL7	I	SINA
							BL8	I	SINA
							FH3	I	SINA
							GUI3A	M	SINA
							OUT	I	SINA
							VT	M	SINA
SINGAM	$\sin(\gamma)$	I	See symbol		/STATE3/(	688)	BL4	I	SINGAM
							BL7	I	SINGAM
							BL8	I	SINGAM
							DER3A	I	SINGAM
							EQUA3	O	SINGAM
							MODELA	I	SINGAM
							MODELB	I	SINGAM
							PDBC	I	SINGAM
							PDY3A	I	SINGAM
							SDER3	I	SINGAM
SINPSI	$\sin(\psi)$	I	See symbol		/STATE3/(	704)	BL4	I	SINPSI
							BL7	I	SINPSI
							BL8	I	SINPSI
							DER3A	I	SINPSI
							EQUA3	O	SINPSI
							MODELA	I	SINPSI
							MODELB	I	SINPSI
							PDBC	I	SINPSI
							PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol		/STATE3/(	706)	BL4	I	SINRHO
							BL7	I	SINRHO
							BL8	I	SINRHO
							DER3A	I	SINRHO
							EQUA3	O	SINRHO
							MODELA	I	SINRHO
							MODELB	I	SINRHO
							OUT	I	SINRHO
							PDBC	I	SINRHO
							PDY3A	I	SINRHO
SIN2RO	$\sin(2\rho)$	M	See symbol		/STATE3/(	755)	BL4	I	SIN2RO
							BL7	M	SIN2RO
							BL8	M	SIN2RO
							MODELA	O	SIN2RO
							MODELB	O	SIN2RO
T	T	I	Thrust	(LBS)	/GENF /(	411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
V	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V
XK3		O	Third control vector governing equation value Corresponds to error in algebraic equation involving α.	/GENF /(	574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3
XK3A		O	Partial of governing equation wrt state or control vector component	/GENF /(	583)	BLGCON	I	XK3A
						BL2	O	XK3A
						BL3	O	XK3A
						BL4	O	XK3A
						BL5	M	XK3A
						BL6	O	XK3A
						BL7	O	XK3A
						BL8	O	XK3A
XK3D		O	Partial of governing equation wrt state or control vector component	/GENF /(	580)	BLGCON	I	XK3D
						BL4	O	XK3D
						BL6	O	XK3D
						BL7	O	XK3D
						BL8	O	XK3D
XK3G		O	Partial of governing equation wrt state or control vector component	/GENF /(	589)	BL4	O	XK3G
						BL7	O	XK3G
						BL8	O	XK3G
XK3M		O	Partial of governing equation wrt state or control vector component	/GENF /(	604)	BL4	O	XK3M
						BL5	M	XK3M
						BL6	O	XK3M
						BL7	O	XK3M
						BL8	O	XK3M
XK3O		O	Partial of governing equation wrt state or control vector component	/GENF /(	598)	BL4	O	XK3O
						BL7	O	XK3O
						BL8	O	XK3O
XK3P		O	Partial of governing equation wrt state or control vector component	/GENF /(	592)	BL4	O	XK3P
						BL7	O	XK3P
						BL8	O	XK3P
XK3R		O	Partial of governing equation wrt state or control vector component	/GENF /(	595)	BL3	O	XK3R
						BL4	O	XK3R
						BL5	M	XK3R
						BL6	O	XK3R
						BL7	O	XK3R
						BL8	O	XK3R

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
XK3T		0	Partial of governing equation wrt state or control vector component	/GENF	/(577)	BLGCON	I	XK3T
						BL4	0	XK3T
						BL6	0	XK3T
						BL7	0	XK3T
						BL8	0	XK3T
XK3V		0	Partial of governing equation wrt state or control vector component	/GENF	/(586)	BL3	0	XK3V
						BL4	0	XK3V
						BL5	M	XK3V
						BL6	0	XK3V
						BL7	0	XK3V
						BL8	0	XK3V

```
      SUBROUTINE BLT                                                      BL7
      COMMON/GENF/                                                        GENF
     *OMG(20)    ,OMSP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R          ,RE        ,MACH      ,PA        ,RD        ,CS        , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP       , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                    GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED     ,IRATED    ,                                             GENF
     *P1         ,P2        ,P3        ,XK1       ,XK2       ,XK3       , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                  MACH, MACHR,   GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                    FRAT
      DIMENSION TPH1(10),TST1(10)                                         GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                  GENF
      COMMON/AECO3/                                                       AECO3
     *APHO       ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       , AECO3
     *SINA       ,COSA      ,PHID      ,PHID      ,PHI       ,SINPHI    , AECO3
     *COSPHI     ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       , AECO3
     *CDOM       ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      , AECO3
     *CM         ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       , AECO3
     *CLAM       ,CL        ,CLA       ,CLM       ,                       AECO3
     *CD         ,CDA       ,CDM                                          AECO3
      COMMON/STATE3/                                                      STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10)   , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBF(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,DCORHO    ,DCORO2    , STATE3D
     *SVBV    (9),OMEGA     ,OMEGA2    ,                                  STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                                 STATE3D
      COMMON/STATE3/                                                      STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                       STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,    EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2),   EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),   EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)              EQUV3
      REAL M,MU,MD                                                        EQUV3
      COMMON/SPECO / UMU2,RORR                                            BL7
C          FLIGHT MODE CONSTRAINT                                         BL7
C                                                                         BL7
      ENTRY BL7010                                                        BL7
      ASSIGN 401 TO IGO                                                   BL7
      ASSIGN 30 TO IAM                                                    BL7
      GO TO 301                                                           BL7   301
      ENTRY BL7001                                                        BL7
      ASSIGN 50 TO IGO                                                    BL7
      GO TO  301                                                          BL7   301
      ENTRY BL7000                                                        BL7
      ASSIGN 60 TO IGO                                                    BL7
C                                                                         BL7
  301 RDOT  = V* SINGAM                                                   BL7
      VDOT = R*OMEGA2*COSRHO*(COSRHO*SINGAM - SINRHO*COSPSI*COSGAM)       BL7
```

```
     * - G*SINGAM + (T*CODAE - DB*COSA - DRAG)/M                         BLT
C                                                                         BLT
C         PRELIMINARY CALC  ENTRY 50 ,40 , 10, 20                         BLT
C                                                                         BLT
  501 ROVRM = RO/M                                                        BLT
      TORO  = 2.* RO                                                      BLT
      TOROR = 2.* ROR                                                     BLT
      TORORR= 2.* RORR                                                    BLT
      GO TO IGO                                                           BLT
C                                                                         BLT
C         PRELIMINARY CALC ENTRY 30 ,10                                   BLT
C                                                                         BLT
  401 CONTINUE                                                            BLT
      SIN2RO = 2.*SINRHO*COSRHO                                           BLT
      COS2RO =(COSRHO-SINRHO)*(COSRHO+SINRHO)                             BLT
      CSRHO = SIN2RO/2.                                                   BLT
      CPSGAM= COSPSI* COSGAM                                              BLT
      CPSSGA= COSPSI* SINGAM                                              BLT
      SRHO2 =SINRHO*SINRHO                                                BLT
      CRHO2 = COSRHO* COSRHO                                              BLT
      ROMEG = R* OMEGA                                                    BLT
      ROMEG2= R* OMEGA2                                                   BLT
      CROGAM= COSRHO* COSGAM                                              BLT
      VDOTR = OMEGA2* COSRHO* ( COSRHO* SINGAM- SINRHO* CPSGAM)           BLT
     *              - GH* SINGAM        - ( DBR* COSA+ DRAGR )/ M         BLT
      VDOTV = -DRAGV/ M                                                   BLT
      RDOTV = SINGAM                                                      BLT
      RDOTG = V* COSGAM                                                   BLT
      VDOTG = ROMEG2*     COSRHO* ( CROGAM         + SINRHO* CPSSGA)      BLT
     *                  - G* COSGAM                                       BLT
      VDOTP =     ROMEG2*  CSRHO*            COSGAM* SINPSI               BLT
      VDOTO =    -ROMEG2* ( SIN2RO*SINGAM + COS2RO*CPSGAM)                BLT
      VDOTM = -( T* CODAE- DB* COSA- DRAG )/ M/ M                         BLT
      GO TO IAM                                                           BLT
C                                                                         BLT
C                                                                         BLT
   30 XK3R=V*RDOT*RORR+TOROR*VDOT +TORO*VDOTR                             BLT
      XK3V  =    TOROR* RDOT- TORO/ M* DRAGV                              BLT
      XK3G  = V* ROR* RDOTG+ TORO* VDOTG                                  BLT
      XK3P  = TORO* VDOTP                                                 BLT
      XK3O = TORO*VDOTO                                                   BLT
      XK3M  = TORO* VDOTM                                                 BLT
C                                                                         BLT
C                                                                         BLT
   50 XK3T  = 2.* ROVRM* CODAE                                            BLT
      XK3D  = 2 * ROVRM* T* SIDAE                                         BLT
      XK3A  = 2 * ROVRM* ( DB* SINA- T* SIDAE- DRAGA )                    BLT
C                                                                         BLT
C         CONSTRAINT EVALUATION                                           BLT
C                                                                         BLT
   60 XK3   = V* RDOT* ROR+ 2.* RO* VDOT                                  BLT
      RETURN                                                              BLT
      END                                                                 BLT
```

20 OCT 72 G 01-46

SUBRØUTINE BL8

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
CODAE	$\cos(\alpha-\delta_E)$	I	See symbol	/GENF /(	549)	ACCEL	I	CODAE
						BL4	I	CODAE
						BL6	I	CODAE
						BL7	I	CODAE
						BL8	I	CODAE
						FH3	I	CODAE
						SDER3	I	CODAE
						VT	O	CODAE
COSA	$\cos\alpha$	I	See symbol	/AECO3 /(	8)	ACCEL	I	COSA
						BL4	I	COSA
						BL6	I	COSA
						BL7	I	COSA
						BL8	I	COSA
						FH3	I	COSA
						OUT	I	COSA
						VT	M	COSA
COSGAM	$\cos(\gamma)$	I	See symbol	/STATE3/(	687)	ACCEL	I	COSGAM
						BL4	I	COSGAM
						BL8	I	COSGAM
						DER3A	I	COSGAM
						EQUA3	O	COSGAM
						MODELA	I	COSGAM
						MODELB	I	COSGAM
						OUT	I	COSGAM
						PDBC	I	COSGAM
						PDY3A	I	COSGAM
COSPSI	$\cos(\psi)$	I	See symbol	/STATE3/(	705)	BL4	I	COSPSI
						BL7	I	COSPSI
						BL8	I	COSPSI
						DER3A	I	COSPSI
						EQUA3	O	COSPSI
						MODELA	I	COSPSI
						MODELB	I	COSPSI
						PDBC	I	COSPSI
						PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol	/STATE3/(	707)	BL4	I	COSRHO
						BL7	I	COSRHO
						BL8	I	COSRHO
						DER3A	I	COSRHO
						EQUA3	M	COSRHO
						MODELA	I	COSRHO
						MODELB	I	COSRHO
						OUT	I	COSRHO
						PDBC	I	COSRHO
						PDY3A	I	COSRHO
COS2RO	$\cos(2\rho)$	M	See symbol	/STATE3/(	756)	BL4	I	COS2RO
						BL7	M	COS2RO
						BL8	M	COS2RO
						MODELA	O	COS2RO
						MODELB	O	COS2RO
DB	D_b	I	Base drag (LBS)	/GENF /(	537)	ACCEL	I	DB
						BL4	I	DB
						BL6	I	DB
						BL7	I	DB
						BL8	I	DB
						EQUA3	I	DB
						FH3	I	DB
						OUT	I	DB
						SDER3	I	DB
						VT	I	DB
DBR		I	Partial of base drag wrt altitude	/GENF /(	536)	ACCEL	I	DBR
						BL4	I	DBR
						BL6	I	DBR
						BL7	I	DBR
						BL8	I	DBR
						EQUA3	I	DBR
						FH3	I	DBR
						VT	I	DBR

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DRAG	D	I	Aerodynamic drag (LBS)	/GENF /(	497)	ACCEL	I	DRAG
						BL5	I	DRAG
						BL7	I	DRAG
						BL8	I	DRAG
						ENVPRM	I	DRAG
						FH3	I	DRAG
						OUT	I	DRAG
						PROPB	O	DRAG
						PROPIN	O	DRAG
						SDER3	I	DRAG
						VT	M	DRAG
DRAGA		I	Partial of drag wrt angle of attack	/GENF /(	534)	ACCEL	I	DRAGA
						BL5	I	DRAGA
						BL7	I	DRAGA
						BL8	I	DRAGA
						FH3	I	DRAGA
						VT	M	DRAGA
DRAGR		I	Partial drag wrt altitude	/GENF /(	533)	ACCEL	I	DRAGR
						BL5	I	DRAGR
						BL7	I	DRAGR
						BL8	I	DRAGR
						FH3	I	DRAGR
						VT	M	DRAGR
DRAGV		I	Partial of drag wrt velocity	/GENF /(	532)	ACCEL	I	DRAGV
						BL5	I	DRAGV
						BL7	I	DRAGV
						BL8	I	DRAGV
						FH3	I	DRAGV
						VT	M	DRAGV
G	g	I	Gravitational attraction (FT/SEC**2)	/GENF /(	301)	BL4	I	G
						BL7	I	G
						BL8	I	G
						DER3A	I	G
						EQUA3	M	G
						MODELA	I	G
						MODELB	I	G
						PDY3A	I	G
						SDER3	I	G
						SDINP	M	G
GH		I	Partial of gravity wrt altitude	/GENF /(	563)	BL7	I	GH
						BL8	I	GH
						EQUA3	O	GH
						PDY3A	I	DGDH
M	m	I	Mass	/STATE3/(	4)	ACCEL	I	M
						BL4	I	M
						BL8	I	M
						EQUA3	I	M
						OUT	I	M
						SDER3	I	M
OMEGA2	ω^2	I	See symbol	/STATE3/(	720)	BL4	I	OMEGA2
						BL7	I	OMEGA2
						BL8	I	OMEGA2
						TOPM	O	OMEGA2
R	R	I	Radial distance from earth center to vehicle (FT)	/GENF /(	305)	BL4	I	R
						BL7	I	R
						BL8	I	R
						DER3A	I	R
						EQUA3	M	R
						MODELA	I	R
						MODELB	I	R
						PDBC	I	R
						PDY3A	I	R
						TRTOSZ	I	R

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
RO	ρ_*	I	Atmospheric density (SLUGS/FT**3)	/GENF /(	309)	BL7	I	RO
						BL8	I	RO
						DER3A	I	RO
						EQUA3	I	RO
						OUT	I	RO
						PDBC	I	RO
						PDY3A	I	RO
ROR		I	Deriv Of density wrt alt.	/GENF /(	313)	BL7	I	ROR
						BL8	I	ROR
						EQUA3	I	ROR
						PDBC	I	ROR
						PDY3A	I	ROR
SIDAE	$\sin(\alpha-\delta_E)$	I	See symbol	/GENF /(	557)	ACCEL	I	SIDAE
						BL4	I	SIDAE
						BL6	I	SIDAE
						BL7	I	SIDAE
						BL8	I	SIDAE
						FH3	I	SIDAE
						VT	O	SIDAE
SINA	$\sin\alpha$	I	See symbol	/AECO3 /(	7)	ACCEL	I	SINA
						BL4	I	SINA
						BL6	I	SINA
						BL7	I	SINA
						BL8	I	SINA
						FH3	I	SINA
						GUI3A	M	SINA
						OUT	I	SINA
						VT	M	SINA
SINGAM	$\sin(\gamma)$	I	See symbol	/STATE3/(	688)	BL4	I	SINGAM
						BL7	I	SINGAM
						BL8	I	SINGAM
						DER3A	I	SINGAM
						EQUA3	O	SINGAM
						MODELA	I	SINGAM
						MODELB	I	SINGAM
						PDBC	I	SINGAM
						PDY3A	I	SINGAM
						SDER3	I	SINGAM
SINPSI	$\sin(\psi)$	I	See symbol	/STATE3/(	704)	BL4	I	SINPSI
						BL7	I	SINPSI
						BL8	I	SINPSI
						DER3A	I	SINPSI
						EQUA3	O	SINPSI
						MODELA	I	SINPSI
						MODELB	I	SINPSI
						PDBC	I	SINPSI
						PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol	/STATE3/(	706)	BL4	I	SINRHO
						BL7	I	SINRHO
						BL8	I	SINRHO
						DER3A	I	SINRHO
						EQUA3	O	SINRHO
						MODELA	I	SINRHO
						MODELB	I	SINRHO
						OUT	I	SINRHO
						PDBC	I	SINRHO
						PDY3A	I	SINRHO
SIN2RO	$\sin(2\rho)$	M	See symbol	/STATE3/(	755)	BL4	I	SIN2RO
						BL7	M	SIN2RO
						BL8	M	SIN2RO
						MODELA	O	SIN2RO
						MODELB	O	SIN2RO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
T	T	I	Thrust (LBS)	/GENF /(	411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROFIN	O	T
						REU3	O	T
						SDER3	I	T
V	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	D	KWOW
						YREF3	M	V
XK3		O	Third control vector governing equation value. Corresponds to error in algebraic equation involving α.	/GENF /(	574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3
XK3A		O	Partial of governing equation wrt state or control vector component	/GENF /(	583)	BLGCON	I	XK3A
						BL2	O	XK3A
						BL3	O	XK3A
						BL4	O	XK3A
						BL5	M	XK3A
						BL6	O	XK3A
						BL7	O	XK3A
						BL8	O	XK3A
XK3D		O	Partial of governing equation wrt state or control vector component	/GENF /(	580)	BLGCON	I	XK3D
						BL4	O	XK3D
						BL6	O	XK3D
						BL7	O	XK3D
						BL8	O	XK3D
XK3G		O	Partial of governing equation wrt state or control vector component	/GENF /(	589)	BL4	O	XK3G
						BL7	O	XK3G
						BL8	O	XK3G

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
XK3M		O	Partial of governing equation wrt state or control vector component	/GENF	/(604)	BL4	O	XK3M
						BL5	M	XK3M
						BL6	O	XK3M
						BL7	O	XK3M
						BL8	O	XK3M
XK3O		O	Partial of governing equation wrt state or control vector component	/GENF	/(598)	BL4	O	XK3O
						BL7	O	XK3O
						BL8	O	XK3O
XK3P		O	Partial of governing equation wrt state or control vector component	/GENF	/(592)	BL4	O	XK3P
						BL7	O	XK3P
						BL8	O	XK3P
XK3R		O	Partial of governing equation wrt state or control vector component	/GENF	/(595)	BL3	O	XK3R
						BL4	O	XK3R
						BL5	M	XK3R
						BL6	O	XK3R
						BL7	O	XK3R
						BL8	O	XK3R
XK3T		O	Partial of governing equation wrt state or control vector component	/GENF	/(577)	BLGCON	I	XK3T
						BL4	O	XK3T
						BL6	O	XK3T
						BL7	O	XK3T
						BL8	O	XK3T
XK3V		O	Partial of governing equation wrt state or control vector component	/GENF	/(586)	BL3	O	XK3V
						BL4	O	XK3V
						BL5	M	XK3V
						BL6	O	XK3V
						BL7	O	XK3V
						BL8	O	XK3V

```
      SUBROUTINE BL8                                                       BL8
      COMMON/GENF/                                                         GENF
     *OMG(20)   ,OMGP(20,2),VARD(9)  ,TOL(9)   ,SVAR(10) ,WDC(20)   ,  GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)  ,COTI(9,9) ,DCON(9) ,DTP       ,  GENF
     *DT5       ,DT        ,G        ,DPSQ     ,Q        ,QS        ,  GENF
     *R         ,RE        ,MACH     ,PA       ,RO       ,CS        ,  GENF
     *VNU       ,PAR       ,ROR      ,CSR      ,VNR      ,SUMSQ     ,  GENF
     *SVSQ      ,TIMEPH    ,TIMES    ,TOP      ,TOS      ,TR(9)     ,  GENF
     *TST(20)   ,TPH   (20),DIS(20)  ,DIP(20)  ,T        ,W         ,  GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20) ,DIS1(20) ,TIME     ,OMP       ,  GENF
     *TIMPR     ,LIFT      ,DRAG     ,TAX      ,TBURN    ,TBU(20)   ,  GENF
     *AE        ,FP        ,FPOLD    ,FPD      ,MACHR    ,MACHV     ,  GENF
     *QR        ,QV        ,FVAC     ,LIFTV    ,                       GENF
     *LIFTR     ,LIFTA     ,          DRAGV    ,DRAGR    ,DRAGA     ,  GENF
     *           LIFTM     ,DBR      ,DB       ,ISP      ,ISPF      ,  GENF
     *                      ULFT     ,ULFTV    ,ULFTR    ,ULFTA     ,  GENF
     *XMCG      ,XMCGV     ,XMCGR    ,XMCGA    ,XMCGM    ,CODAE     ,  GENF
     *CULFT     ,CT        ,CALPHA   ,CDE      ,DELTAE   ,SID       ,  GENF
     *COD       ,SIDAE     ,XCG      ,ZCG      ,XJ                     GENF
      COMMON / GENF  /                                                     GENF
     *XJV       ,XJR       ,GH       ,GAMMAD   ,XKG      ,XKP       ,  GENF
     *FRATED    ,IRATED    ,                                             GENF
     *P1        ,P2        ,P3      ,XK1       ,XK2      ,XK3       ,  GENF
     *XK1T      ,XK2T      ,XK3T     ,XK1D     ,XK2D     ,XK3D      ,  GENF
     *XK1A      ,XK2A      ,XK3A     ,XK1V     ,XK2V     ,XK3V      ,  GENF
     *XK1G      ,XK2G      ,XK3G     ,XK1P     ,XK2P     ,XK3P      ,  GENF
     *XK1R      ,XK2R      ,XK3R     ,XK1O     ,XK2O     ,XK3O      ,  GENF
     *XK1U      ,XK2U      ,XK3U     ,XK1M     ,XK2M     ,XK3M      ,  GENF
     *PV        ,PG        ,PP       ,PR       ,PO       ,DPDY(3,8)    GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                     FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      COMMON/STATE3/                                                       STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)    ,SVY(10)   ,  STATE3D
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM   ,SAVBP(15) ,  STATE3D
     *SINPSI    ,COSPSI    ,SINRHO   ,COSRHO   ,OCORHO   ,OCORO2    ,  STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2   ,                                   STATE3D
     *VDV       ,GDV       ,RDV      ,MDV      ,PDV      ,ODV       ,  STATE3D
     *UDV       ,VDG       ,GDG      ,RDG      ,PDG      ,ODG       ,  STATE3D
     *UDG       ,VDR       ,GDR      ,MDR      ,PDR      ,ODR       ,  STATE3D
     *UDR       ,VDM       ,GDM      ,MDM      ,PDM      ,VDP       ,  STATE3D
     *GDP       ,PDP       ,ODP      ,UDP      ,VDO      ,GDO       ,  STATE3D
     *PDO       ,UDO       ,HTDV     ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                  STATE3D
      COMMON/STATE3/                                                       STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                         STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,     EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2),    EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),    EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)               EQUV3
      REAL M,MU,MD                                                         EQUV3
      COMMON/AECO3/                                                        AECO3
     *APHO      ,APHR      ,ALPHA    ,VDA      ,GDA      ,PDA       ,  AECO3
     *SINA      ,COSA      ,PHIO     ,PHID     ,PHI      ,SINPHI    ,  AECO3
     *COSPHI    ,GDPH      ,PDPH     ,XLAMA(9) ,XLAMP(9) ,CDO       ,  AECO3
     *CDOM      ,CLO       ,FK       ,XCGM     ,ZCGM     ,CLOM      ,  AECO3
     *CM        ,CMA       ,CMAM ,CMM ,CMO     ,CMOM     ,FKM       ,  AECO3
     *CLAM      ,CL        ,CLA      ,CLM      ,                         AECO3
     *CD        ,CDA       ,CDM                                          AECO3
      COMMON /SPECO/ UMU2,RORR                                             BL8
      ENTRY BL8010                                                         BL8
      ASSIGN 301 TO IAM                                                    BL8
      ASSIGN 30 TO IGO                                                     BL8
      GO TO 601                                                            BL8    601
      ENTRY BL8001                                                         BL8
      ASSIGN 501 TO IAM                                                    BL8
       ASSIGN 50 TO IGO                                                    BL8
      GO  TO 601                                                           BL8    601
      ENTRY BL8000                                                         BL8
      ASSIGN 60 TO IAM                                                     BL8
C                                                                          BL8
  601 ROOT    = V* SINGAM                                                  BL8
      R6R     = ROR/6.3                                                    BL8
      R6RR    = RORR/6.3                                                   BL8
```

```
      V115   = V**1.15                                                  BL8
      SIN2RO = 2 *SINRHO*COSRHO                                         BL8
      COS2RO = (COSRHO -SINRHO)*(COSRHO +SINRHO)                        BL8
      CSRHO  = SIN2RO /2.                                               BL8
      CPSGAM=COSPSI*COSGAM                                              BL8
      CPSSGA = COSPSI* SINGAM                                           BL8
      ROMEG2 = R* OMEGA2                                                BL8
      CROGAM = COSRHO* COSGAM                                           BL8
      CROSGA = COSRHO* SINGAM                                           BL8
      VDOT   = ROMEG2   * COSRHO*( CROSGA-         SINRHO* CPSGAM)      BL8
     1                - G* SINGAM+ (T* CODAE- DB* COSA- DRAG)/ M        BL8
      GO TO IAM                                                         BL8
C                                                                       BL8
C          PRELIMINARY CALCULATIONS FOR ENTRY 20                        BL8
C                                                                       BL8
  301 VDOTR=OMEGA2*COSRHO*(CROSGA-SINRHO*CPSGAM)-GH*SINGAM-(DBR*COSA    BL8
     *  +DRAGR)/M                                                       BL8
      RDOTV  = SINGAM                                                   BL8
      VDOTV  =-DRAGV/ M                                                 BL8
      RDOTG  = V* COSGAM                                                BL8
      VDOTG  = ROMEG2* COSRHO*( CROGAM+ SINRHO* CPSSGA)- G* COSGAM      BL8
      VDOTP  = ROMEG2* CSRHO * SINPSI* COSGAM                           BL8
      VDOTO  =-ROMEG2*( 2 *CSRHO * SINGAM + COS2RO* CPSGAM)             BL8
      VDOTM  =-(T*CODAE - DB* COSA - DRAG)/M/M                          BL8
C                                                                       BL8
C          PRELIMINARY CALCULATIONS FOR ENTRY 40                        BL8
C                                                                       BL8
  501 VDOTT=CODAE/M                                                     BL8
      VDOTD=T*SIDAE/M                                                   BL8
      VDOTA=-VDOTD+(DB*SINA-DRAGA)/M                                    BL8
      GO TO IGO                                                         BL8
   30 XK3R=RDR*VDOT+RO*VDOTR + V115*R6RR*RDOT                           BL8
      XK3V   = V115*R6R*2.15*SINGAM+RO*VDOTV                            BL8
      XK3G   =  V115*R6R*RDOTG + RO*VDOTG                               BL8
      XK3P   = RO* VDOTP                                                BL8
      XK3O   = RO* VDOTO                                                BL8
      XK3M   = RO* VDOTM                                                BL8
C                                                                       BL8
C                                                                       BL8
   50 XK3T=RO*VDOTT                                                     BL8
      XK3D   = RO* VDOTD                                                BL8
      XK3A   = RO* VDOTA                                                BL8
C                                                                       BL8
C          CALCULATE CONSTRAINT                                         BL8
C                                                                       BL8
   60 XK3    = V115*RDOT*R6R + RO*VDOT                                  BL8
      RETURN                                                            BL8
      END                                                               BL8
```

20 OCT 72 G 01-46

SUBRØUTINE BNTG

BNTG

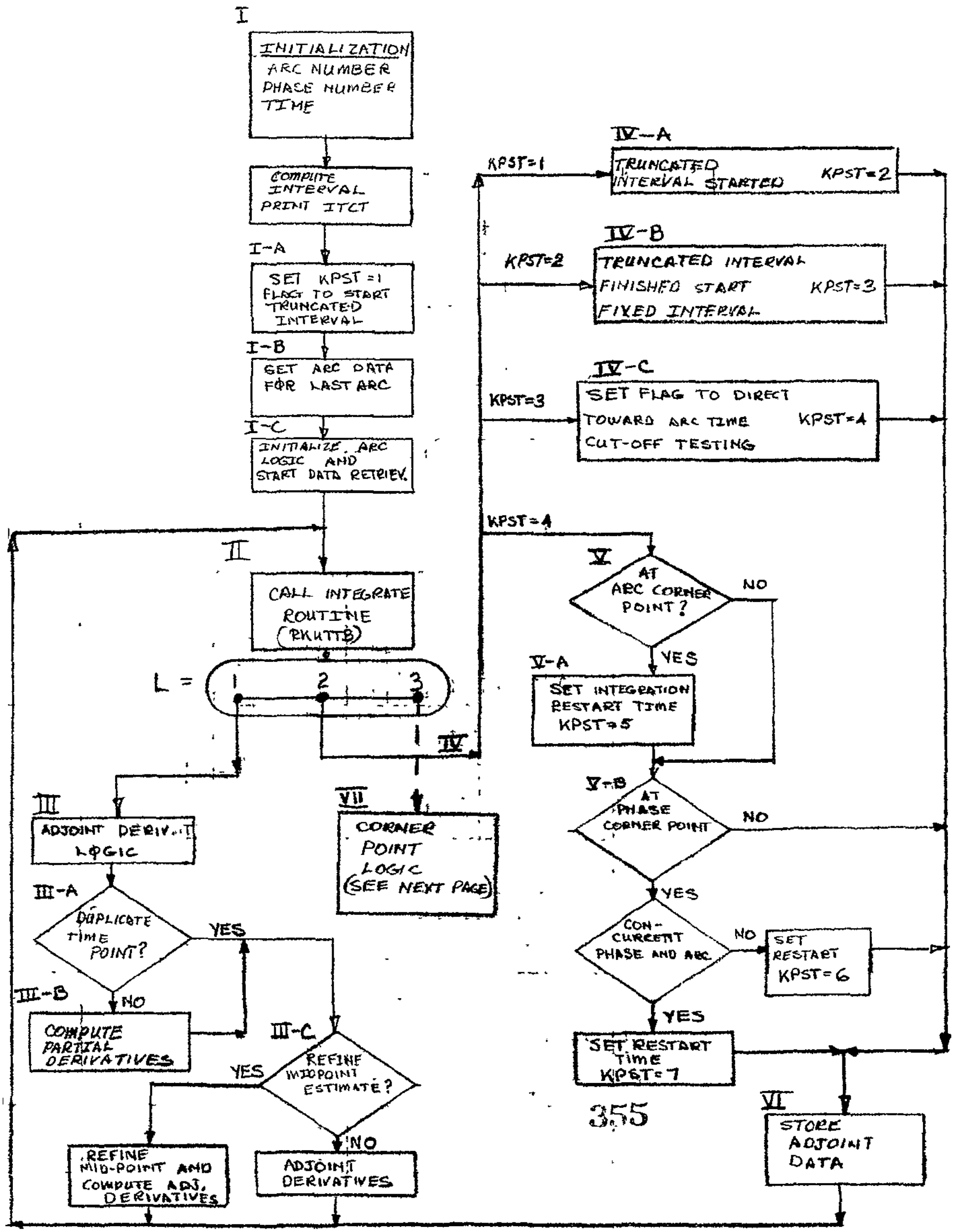
Purpose

BNTG is an executive routine that controls integration of adjoint system. On the CDC computer overlay software, it is a program that heads up an overlay, whereas on the UNIVAC system it is a subroutine.

Description

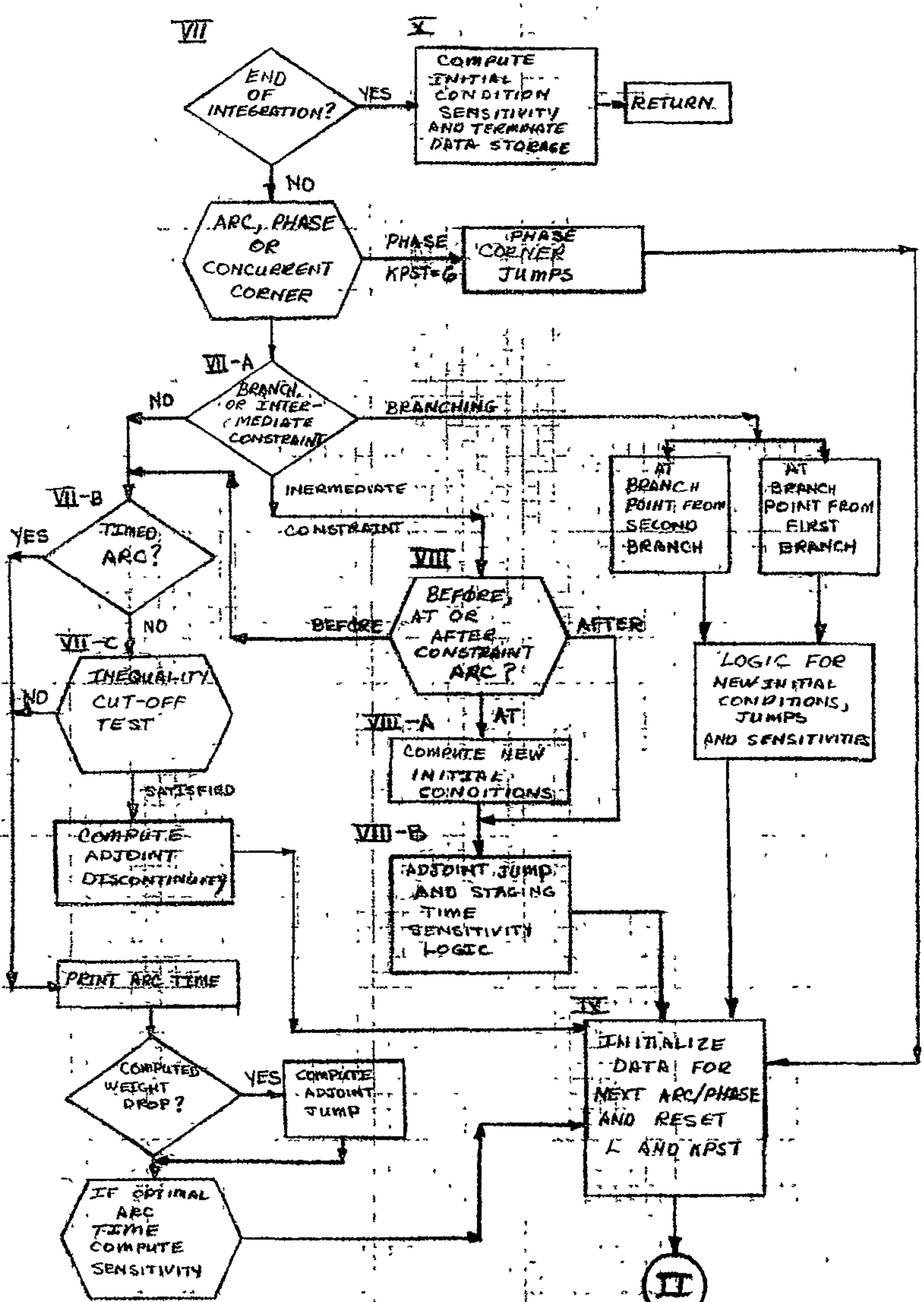
The key aspect of the backward adjoint integration logic is that the compute intervals are precisely matched to the previous nominal forward trajectory compute intervals. The integration is started at the end of each arc using a saved truncated time interval (the same truncated interval required to satisfy the arc cut-off on the nominal trajectory). After completing the interval, the integration is restarted with the full step size and continues until the beginning of the arc. This process continues until the beginning of the first arc where the logic recognizes (VII) that initial time has been reached and proceeds to terminate storage of adjoint data (X).

BNTG


I
INITIALIZATION
ARC NUMBER
PHASE NUMBER
TIME
COMPUTE INTERVAL
PRINT ITCT
I-A
SET KPST =1
FLAG TO START TRUNCATED INTERVAL
I-B
GET ARC DATA FOR LAST ARC
I-C
INITIALIZE ARC LOGIC AND START DATA RETRIEV.
II
CALL INTEGRATE ROUTINE (RKUTTB)
L = 1 2 3
III
ADJOINT DERIV. LOGIC
III-A
DUPLICATE TIME POINT?
YES
NO
III-B
COMPUTE PARTIAL DERIVATIVES
III-C
REFINE MIDPOINT ESTIMATE?
YES
NO
REFINE MID-POINT AND COMPUTE ADJ. DERIVATIVES
ADJOINT DERIVATIVES
VII
CORNER POINT LOGIC (SEE NEXT PAGE)
IV
KPST=1
IV-A
TRUNCATED INTERVAL STARTED
KPST=2
KPST=2
IV-B
TRUNCATED INTERVAL FINISHED START FIXED INTERVAL
KPST=3
KPST=3
IV-C
SET FLAG TO DIRECT TOWARD ARC TIME CUT-OFF TESTING
KPST=4
KPST=4
V
AT ARC CORNER POINT?
NO
YES
V-A
SET INTEGRATION RESTART TIME
KPST=5
V-B
AT PHASE CORNER POINT
NO
YES
CON-CURRENT PHASE AND ARC
NO
SET RESTART
KPST=6
YES
SET RESTART TIME
KPST=7
VI
STORE ADJOINT DATA

350

BNTG (CONTINUED)



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
A	A	I	Control integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BSTO3	I	A
						MTX3A	I	A
						PAYO2	I	A
						SDINP	I	A
						TRAN3	I	A
ARCDA	S_{ref}	I	Aerodynamic reference area (FT^2)	/ARCDAT/(	1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SDINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF
DIP		I	Array of phase end integration intervals for trial trajectory	/GENF /(	391)	BNTG	I	DIP
						FNTG	O	DIP
DIS		I	Array of arc end integration intervals for trial trajectory	/GENF /(	371)	BNTG	I	DIS
						FNTG	O	DIS
DT		M	Integration interval (SEC)	/GENF /(	300)	BNTG	M	DT
						FNTG	M	DT
						REU3	I	DT
						RKTA3A	I	P
						RKTB3A	I	P
						STP3	I	DT
						YREF3	O	DT
DTNC	$\Delta\tau$	I	Integration interval (SEC)	/ARCDAT/(	5)	BNTG	I	DTNC
						FNTG	I	DTNC
						GEINP	M	DTNC
						PROPIN	I	DTNC
IARC		M	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
INTB		I	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
						BNTG	I	INTB
						ENVPRM	I	INTB
						FNTG	I	INTB
						SDINP	M	INTB
						TEST	I	INTB
						TRAN3	I	INTB
						TRTOSZ	I	INTB
IPH		M	Phase number	/XCODES/(	143)	ADID3A	I	IPH
						ADJUST	I	IPH
						AST3	I	IPH
						BNTG	M	IPH
						FNTG	M	IPH
						GETIT	I	IPH
						GUI3A	I	IPH
						SDINP	M	IPH
IPP		W	Phase number at last derivative evaluation	/BNTG /(*	)	BNTG	W	IPP

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ITCT		I	Iteration counter	/XCODES/(	148)	BNTG	I	ITCT
						OUT	I	ITCT
						TEST	M	ITCT
						TOPM	M	ITCT
ITT		W	Arc number at last derivative evaluation	/BNTG /(*	)	BNTG	W	ITT
JGID		I	Control option flag array	/XCODES/(	32)	BNTG	I	JGID
						FNTG	I	JGID
						SDINP	M	JGID
JGII		O	Control option	/XCODES/(	195)	ACCEL	I	JGII
						BNTG	O	JGII
						DER3A	I	JGII
						FNTG	M	JGII
						GUI3A	I	JGII
						MODELA	I	JGII
						MODELB	I	JGII
						MTX3A	I	JGII
						PDY3A	I	JGII
JK		I	Integration routine flag tells which derivative evaluation in Runge-Kutta cycle	/XCODES/(	151)	ADIC3A	M	JK
						BNTG	I	JK
						MODELA	I	JK
						PAY02	M	JK
						RKTA3A	M	J
						RKTB3A	M	J
JPH		I	Phase cut-off option flag	/XCODES/(	72)	BNTG	I	JPH
						FNTG	M	JPH
						SDINP	M	JPH
JPS		M	Absolute value of phase cut-off option code	/XCODES/(	152)	ADID3A	I	JPS
						BNTG	M	JPS
						FNTG	M	JPS
						STP3	I	JPS
						TOL3	I	JPS
JS		M	Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
						ADIC3A	I	JS
						ADID3A	I	JS
						BNTG	M	JS
						FNTG	M	JS
						PROPB	I	JS
						PROPIN	I	JS
						STP3	I	JS
						TOL3	I	JS
JST		I	Arc cut-off option flag	/XCODES/(	112)	ADICB3	I	JST
						BNTG	I	JST
						FNTG	I	JST
						SDINP	M	JST
KPST		M	Controls logic for compute interval during adjoint integration	/XCODES/(	155)	BNTG	M	KPST
						FNTG	M	KPST
L		M	Integration traffic control flag L = 1 means evaluate derivatives = 2 check cut-off = 3 print or cut-off detected	/XCODES/(	177)	BNTG	M	L
						FNTG	M	L
						OUT	I	L
						RKTA3A	M	L
						RKTB3A	M	L
						SDINP	M	L
NB		O	Extent of integration set during adjoints on branch problem	/XCODES/(	179)	ADEQ3A	I	NB
						ADICB3	M	NB
						ADIC3A	M	NB
						BNTG	O	NB
						RKTB3A	I	NB
						STVRL3	I	NB

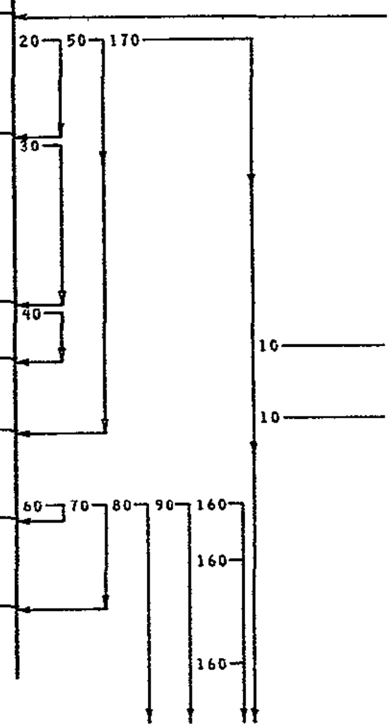
FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
NCN		I	Number of elements in $d\Psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BSTD3	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAYO2	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NICNB		I	Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
						ADIC3A	I	NICNB
						BNTG	I	NICNB
						REU3	I	NICNB
						SDINP	M	NICNB
						TEST	I	NICNB
						TRAN3	I	NICNB
NPARA		I	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	D	NPARA
NPH		I	Number of phases in trajectory	/XCODES/(	164)	BNTG	I	NPH
						FNTG	O	NPH
						PRMSET	I	NPH
						SDINP	M	NPH
						TEST	I	NPH
						TOPM	I	NPH
NSAB		I	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB
NST		I	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
SVAR	$y\|_{t=0}$	I	Array of state values at initial problem time [sd]	/GENF	/(79)	ADJUST	O	SVAR
						BNTG	I	SVAR
						FNTG	I	SVAR
						PRMSET	M	SVAR
						REU3	I	SVAR
						SDINP	M	SVAR
						TEST	I	SVAR
						TOPM	I	SVAR
						TRTOSZ	I	SVAR
TIME	t	M	Time (elapsed)	/GENF	/(493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
TIMPR		O	Trajectory print time	/GENF	/(495)	BNTG	O	TIMPR
						FNTG	M	TIMPR
						RKTA3A	I	TP
						RKTB3A	I	TP
TOP		M	Elasped time at phase initiation	/GENF	/(320)	BNTG	M	TOP
						EQUA3	I	TOP
						FNTG	M	TOP
TOS		M	Elasped time at arc initiation	/GENF	/(321)	BNTG	M	TOS
						EQUA3	I	TOS
						FNTG	M	TOS
TPH1		I	Phase end times for nominal trajectory	/GENF	/(413)	BNTG	I	TPH1
						GETIT	I	TPH1
						SDINP	O	TPH1
						TEST	O	TPH1
						TOPM	I	TPH1
TST		I	Array of arc end times on trial trajectory [sd]	/GENF	/(331)	ADICB3	I	TST
						BNTG	I	TST
						FNTG	O	TST
						TEST	I	TST
TST1		I	Arc end times for nominal trajectory	/GENF	/(433)	BNTG	I	TST1
						GETIT	I	TST1
						PROPIN	I	TST1
						SDINP	O	TST1
						TEST	O	TST1
						TOPM	I	TST1
						TRAN3	I	TST1
						TRTOSZ	I	TST1
TSV		W	Time at last derivative evaluation	/BNTG	/(*)	BNTG	W	TSV
WDC		I	Array of drop weight per arc[sd] (LBS)	/GENF	/(89)	BNTG	I	WDC
						REU3	I	WDC
						SDINP	M	WDC

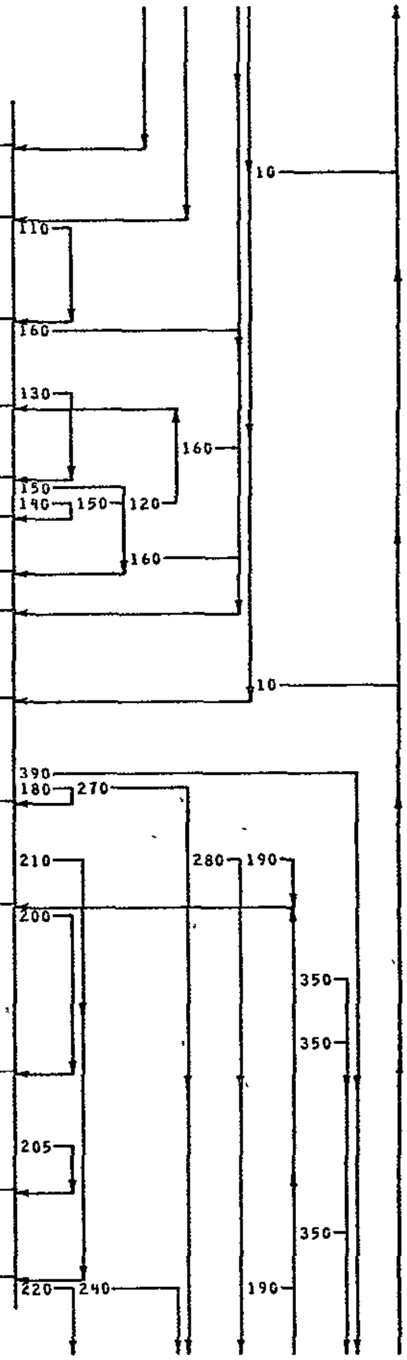
```
      PROGRAM BNTG                                                      BNTG
C                                                                       COMN
C         CONTROLS  INTEGRATION OF  ADJOINT DIFFERENTIAL EQUATIONS      COMN
C         USES  TRAFFIC FLAG  L  FOR THE  RUNGE-KUTTA INTEGRATION(RKUTTB COMN
C         WHERE    L=1  MEANS  COMPUTE  DERIVATIVES                     COMN
C                  L=2         TEST  STOPPING CONDITION                 COMN
C                  L=3         PRINT  OR  CORNER  POINT                 COMN
C                  L=4         RESTART  INTEGRATION                     COMN
C        XPST IS THE  ARC INITIATION AND STAGING FLAG                   COMN
C   =1 FOR STARTING  ARC AT TRUNCATED  COMPUTE INERVAL                  COMN
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI        ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)           ,NCNST      ,NSB        ,NSAB       ,NICNB      , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR       ,IFB        ,IND        , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST       ,IARC       ,ISTART     , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK         ,JPS        ,JS         , XCODES
     *KOP        ,KPST      ,K         ,KST        ,NAD        ,NCASE      , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP        ,NPH        ,N          , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN       ,IPHN       ,ISTNB      , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP      ,ISTPP      ,L          , XCODES
     *IFOB       ,NB        ,LB        ,MB         ,NPHP       ,NPHB       , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGI1,MTT,MPIN(20),JP1,JP2,JP3           XCODES
      COMMON/STS/                                                       STS
     *DPAY       ,PMIN      ,WORK (20),NWDS        ,IPC    (7),NITER   , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)             STS
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF       ,EJ        ,XISP      ,TMULT      ,DTNC       ,DTPI       , ARCDAT
     *IATM       ,IMODE     ,JAER      ,JPRO       ,QMAX       ,GMAX       , ARCDAT
     *XLMAX      ,HDMAX     ,GNDOT     ,ALFMAX     ,PHMAX      ,MAEA       , ARCDAT
     *MAEB       ,MAEC      ,MAED      ,MAEE       ,MAEF       ,MAEG       , ARCDAT
     *MT         ,MISP      ,MXCG      ,MZCG       ,MWDA       ,MWDB       , ARCDAT
     *MDB        ,XCGR      ,ZCGR      ,XE         ,ZE         ,XT         , ARCDAT
     *DREF       ,MCND      ,RHOB      ,QMULT      ,REMAX                    ARCDAT
     *  ,FRATE  ,  ARCD(9)                                              RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      COMMON/GLOBAL/                                                    GLOBAL
     *GR         ,ER        ,OMGZ      ,XLAMRF     ,YMURF      ,LUM        GLOBAL
     *,JJOP(10)  ,IFATAL    ,NARC        ,NBRAN    ,NFARC      ,ID(4)      GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG   ,MAXTAB                                GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)     GLOBAL
     *,ITPSO ,KSOL      ,KGLOBL(8)                                      RETAP
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)     ,SVAR(10)  ,WDC(20)     , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9)  ,DCON(9)    ,DTP        , GENF
     *DTS        ,DT        ,G         ,DPSQ       ,Q          ,QS         , GENF
     *R          ,RE        ,MACH      ,PA         ,RO         ,CS         , GENF
     *VNU        ,PAR       ,ROR       ,CSR        ,VNR        ,SUMSQ      , GENF
     *SVSQ       ,TIMEP..   ,TIMES     ,TOP        ,TOS        ,TR(9)      , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)    ,T          ,W          , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)   ,TIME       ,OMP        , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX        ,TBURN      ,TBU(20)    , GENF
     *AE         ,FP        ,FPOLD     ,FPD        ,MACHR      ,MACHV      , GENF
     *QR         ,QV        ,FVAC      ,LIFTV      ,                         GENF
     *LIFTR      ,LIFTA     ,           DRAGV      ,DRAGR      ,DRAGA      , GENF
     *            LIFTM     ,DBR       ,DB         ,ISP        ,ISPF       , GENF
     *                       ULFT      ,ULFTV      ,ULFTR      ,ULFTA      , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA      ,XMCGM      ,CODAE      , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE        ,DELTAE     ,SID        , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG        ,XJ                       GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD     ,XKG        ,XKP        , GENF
     *FRATED    ,IRATED     ,                                           GENF
     *P1         ,P2        ,P3      ,XK1          ,XK2        ,XK3        , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D       ,XK2D       ,XK3D       , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V       ,XK2V       ,XK3V       , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P       ,XK2P       ,XK3P       , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O       ,XK2O       ,XK3O       , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M       ,XK2M       ,XK3M       , GENF
     *PV         ,PG        ,PP        ,PR         ,PD         ,DPDY(3,8)  , GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                 MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/DATA/                                                      DATA
```

20 OCT 72 6.01-46

```
     *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      , DATA
     *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4        DATA
      COMMON/PARAM/                                                     PARAM
     *IPOINT(12),NPARA,NPA  ,SPARA(9,12),WTPO  (9),WTP   (12),          PARAM
     *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                       PARAM
     *,DELP(9)                                                          PARAM
      EQUIVALENCE(INEQFL(20),INQF)                                      NOS
      EQUIVALENCE (INEQS,ILAB(2))                                       APR27
C   THIS PROGRAM  DOES  THE INTEGRATION  OF  THE ADJOINT  SOLUTION      BNTG
C                                                                       COMN
C          I      INITIALIZATION                                        COMN
      IARC= NST                                                         BNTG
      IPH  = NPH                                                        BNTG
      TIME = TST(IARC)                                                  BNTG
      TOS  =         TST1(IARC)                                         BNTG
      TOP  =         TPH1(IPH)                                          BNTG
C     CALCULATE  COMP INTERVAL                                          BNTG
      DT =-DIS(NST)                                                     BNTG
      CALL IPR(4HITCT,A,ITCT,1,1)                                       BNTG
      KPST =1                                                           BNTG
C          I-A    SET FLAG TO START ARC AT TRUNCATED INTERVAL           COMN
      JS=IABS(JST(NST-1))                                               BNTG
      JGII=JGID(NPH,2)                                                  BNTG
C          I-B    GET  DATA FOR  ARC                                    COMN
      CALL READMS(9,ARCDA,51,IARC)                                      FRAT
      JPS=IABS(JPH(NPH-1 ,2))                                           BNTG
      L=4                                                               BNTG
      TIMPR=-1.E6                                                       BNTG
C          I-C     INITIALIZE  ARC INFOR  AND  STORAGE RETRIEVAL        COMN
      CALL PROPB                                                        PO14
      CALL BEGNA                                                        BNTG
      CALL BEGNB                                                        BNTG
C                                                                       COMN
C         II     INTEGRATION   SUBROUTINE   RKUTTB                      COMN
   10 CALL  RKUTTB                                                      BNTG
      GO  TO (20,50,170),L                                              BNTG
C                                                                       COMN
C          III                                                          COMN
C   CALCULATE ADJOINT  DERIVATIVES                                      BNTG
C          III-A   SKIP DERIV  CALC     AT  DUPLICATE TIME POINT        COMN
C           EXCEPT  AT  STAGE  CHANGE                                   COMN
   20 IF(TIME EQ TSV.AND ITT.EQ.IARC.AND.IPP.EQ IPH) GO TO 30           BNTG
C                                                                       COMN
C          III-B     PARTIAL  DERIVATIVES                               COMN
      CALL MODEL                                                        BNTG
      TSV = TIME                                                        BNTG
      ITT=IARC                                                          BNTG
      IPP =IPH                                                          BNTG
C                                                                       COMN
C          III-C   SPECIAL  ADJOINT DERIVATIVE  WITH REFINED            COMN
C                  ESTIMATE OF  IMPULSE  RESPONSE FUNCTIONS             COMN
   30 IF(JK.NE.3) GO TO 40                                              BNTG
      CALL SNSADJ                                                       BNTG
      GO TO 10                                                          BNTG
   40 CALL ADEQ                                                         BNTG
C                                                                       COMN
C          III-D     ADJOINT    DERIVATIVES   AND A MATRIX INTEGRANDS   COMN
      GO TO 10                                                          BNTG
   50 CONTINUE                                                          BNTG
C                                                                       COMN
C          IV      ***         ***                                      COMN
C     TEST FOR  STAGING  CONDITIONS                                     BNTG
      GO TO (60,70,80,90,160,160,160),KPST                              BNTG
   60 KPST =2                                                           BNTG
C          IV-A     TRUNCATED   INTERVAL   HAS  STARTED                 COMN
      GO  TO   160                                                      BNTG
C                                                                       COMN
C          IV-B     TRUNCATED  INTERVAL FINISHED START FIXED INTERVAL   COMN
   70 KPST =3                                                           BNTG
      L=4                                                               BNTG
      DT =-DTNC                                                         BNTG
      GO TO 160                                                         BNTG
```



```
C                                                                  COMN
C       IV-C     SET FLAG TO DIRECT TOWARD CUT-OFF TESTING         COMN
     80 KPST = 4                                                   BNTG
        GO TO  10                                                  BNTG
C         V         ***                                            COMN
C        NORMAL LOGIC  FOR STAGING FOLLOWS                         BNTG
     90 IF( ABS(TIME +DT-TOS )        GT 1 E-5) GO TO 110          BNTG
C                                                                  COMN
C         V-A     ARC  INITIAL POINT    SET  INTEGRATION  RESTART TIME  COMN
    100 KPST = 5                                                   BNTG
        TIMPR = TOS                                                BNTG
C       CHECK  FOR  PHASING                                        BNTG
    110 IF(ABS(TIME +DT - TOP       ) GT 1.E-5) GO TO 160          BNTG
C                                                                  COMN
C         V-B     PHASE  INITIAL POINT                             COMN
C                 TEST   FOR   CONCURRENT  STAGING                 COMN
        IF(KPST EQ 5)  GO TO 130                                   BNTG
    120 KPST = 6                                                   BNTG
        TIMPR = TOP                                                BNTG
        GO TO  160                                                 BNTG
C       COMBINED   STAGING AND  PHASING                            BNTG
    130 IF (ABS(TOP-TOS).LT.1.E-5) GO TO 150                       BNTG
        IF ( TOP -TOS) 120,150,140                                 BNTG
    140 KPST = 5                                                   BNTG
        TIMPR= TOS                                                 BNTG
        GO TO 160                                                  BNTG
    150 KPST = 7                                                   BNTG
        TIMPR =TOS                                                 BNTG
    160 CONTINUE                                                   BNTG
        CALL BSTO                                                  BNTG
C                                                                  COMN
C         VI     STORE   ADJOINT DATA                              COMN
        GO  TO  10                                                 BNTG
    170 CONTINUE                                                   BNTG
C                                                                  COMN
C        VII    TEST FOR END  OF INTEGRATION  YES GO TO   X        COMN
C               ELSE    RESTART FOR  NEXT ARC  AT  IX              COMN
        IF(ABS(TIME-SVAR(1)).LE.1.E-5)  GO  TO  390                BNTG
        IF(KPST-6) 180,270,180                                     BNTG
    180 CONTINUE                                                   BNTG
C                                                                  COMN
C         VII-A    BRANCHING  AND INTERMEDIATE  CONSTRAINT  TEST   COMN
        IF(INTB-1) 190,210,280                                     BNTG
C       NO BRANCH OR  CONST.                                       BNTG
C         VII-B      TIMED  ARC ,IF SO  SKIP ADJOINT JUMPS         COMN
    190 IF(JS LT 3) GO TO 200                                      BNTG
C         VII-C      ADJOINT  DISCON  JUMP                         COMN
C                IF INEQUALITY CUTTOFF  HAS NOT BEEN REACHED       COMN
C                TREAT AS  A  TIMED ARC ( INEDS FLAG NON-ZERO)     COMN
        IF(INEDS  NE.O) GO TO 350                                  APR27
        CALL ADIDS(1)                                              BNTG
        CALL STVARL(1,NCN)                                         BNTG
C       INT  CONST  LOGIC                                          BNTG
        GO TO  350                                                 BNTG
C           COMUTE  STAGING  TIME  SENSITIVITY                     BNTG
    200 CONTINUE                                                   UH
C                                                                  COMN
C                PRINT  ARC  TIMES                                 COMN
        CALL IPR(4HTIME  ,TIME,I,1,0)                              UH
        IF(WDC(IARC-1) NE -99999999.) GO TO 205                    UH
         CALL ADJUMP(1,NCN)                                        UH
        CALL STVARL(1,NCN)                                         UH
    205 CONTINUE                                                   UH
        IF(NPARA GT 0) CALL STAU(1,NCN)                            PH1SZ
        GO  TO  350                                                BNTG
C                                                                  COMN
C         VIII       INTERMEDIATE   CONSTRAINT  ARC  TEST          COMN
    210 IF( IARC-1-NSB) 190,220,240                                BNTG
```



20 OCT 72 G 01-46

```
      C                                                                 COMN
      C         VIII-A   INTRODUCE AND INITIALIZE NEW  ADJOINT  SET     COMN
        220 CALL ADICB                                                  BNTG
      C         VIII-B    ADJOINT   JUMP + STAGING  TIME  SENSITIVITY  LOGIC COMN
      C            FOR  BOTH  INT. CONST.  AND  BRANCH  TRUNK           COMN
            IF(JS.LT.3) GO TO 230                                       BNTG
            CALL ADIDS(NICNB+1)                                         BNTG
            CALL STVARL(1,NCN)                                          BNTG
            GO TO 350                                                   BNTG
        230 CALL STVARL(1,NCN)                                          BNTG
            IF(NPARA.GT.0) CALL STAU( 1,NCN)                            BNTG
            GO TO 350                                                   BNTG
        240 IF(JS.LT.3) GO TO 260                                       BNTG
            CALL ADIDS( NICNB +1)                                       BNTG
        250 CALL STVARL(NICNB+1,NCN)                                    BNTG
            GO TO 350                                                   BNTG
        260 CALL STVARL(NICNB+1,NCN)                                    BNTG
            IF(NPARA.GT.0)  CALL STAU(NICNB +1,NCN)                     BNTG
            GO TO 350                                                   BNTG
        270 IF(JPS.LT 3) GO TO 360                                      BNTG
            CALL ADIDP( 1)                                              BNTG
            CALL  STVARL(1,NCN)                                         BNTG
            GO TO 360                                                   BNTG
        280 IF( IARC-1-(NSAB+NSB))300,290,240                           BNTG
      C     AT  BRANCHING  POINT  FROM  SEGMENT III                     BNTG
        290 CALL ADI3B                                                  BNTG
            CALL STVARL(1,NICNB)                                        BNTG
            GO  TO 350                                                  BNTG
        300 IF(IARC-1-NSB) 190,310,320                                  BNTG
      C     AT BRANCHING  PT  FROM  SEGMENT  II                         BNTG
        310 NB =1                                                       BNTG
      C  RESTORE BRANCH ELEMENT IN A MATRIX                             BNTG
            CALL ADI2B                                                  BNTG
            GO  TO 190                                                  BNTG
      C   IN   SEGMENT II                                               BNTG
        320 IF(JS.LT.3) GO  TO  340                                     BNTG
            CALL ADIDSS(NICNB)                                          BNTG
        330 CALL STVARL(1,NICNB)                                        BNTG
            GO  TO  350                                                 BNTG
        340 CALL STVARL(1,NICNB)                                        BNTG
            IF(NPARA GT 0)  CALL  STAU(1,NICNB)                         BNTG
      C                                                                 COMN
      C          IX     INITIALIZE  DATA FOR NEXT ARC  AND RESET  L AND KPST COMN
        350 IARC=IARC-1                                                 BNTG
      C     CALCULATE  DT                                               BNTG
            DT =-DIS(IARC)                                              BNTG
            CALL READMS(9,ARCDA,51,IARC)                                FRAT
            JS=IABS(JST(IARC-1))                                        APR27
            CALL PROPB                                                  BNTG
            TOS=TST1(IARC)                                              BNTG
            IF(KPST-6) 380,360,370                                      BNTG
      C   CALCULATE DT                                                  BNTG
        360 DT =-DIP(IPH-1)                                             BNTG
        370 IPH =IPH-1                                                  BNTG
            JPS=IABS(JPH(IPH-1,2))                                      BNTG
            JGII=JGID(IPH,2)                                            BNTG
            TOP=TPH1(IPH)                                               BNTG
        380 CONTINUE                                                    BNTG
            KPST =1                                                     BNTG
            TIMPR = -1.E6                                               BNTG
            L=4                                                         BNTG
            GO TO 10                                                    BNTG
        390 CONTINUE                                                    BNTG
      C                                                                 COMN
      C          X      COMPUTE  IC  SENSITIVITY  AND TERMINATE DATA STORAGE COMN
            IF(NPARA.GT.0) CALL  SINIT(1,NCN)                           BNTG
            CALL ENDB                                                   BNTG
```

20 OCT 72 G 01-46

```
283.       END                                                            SNTG
```

20 OCT 72 6.01-46

SUBRØUTINE
BSTØ3

Subroutine BST03

Purpose

BST03 handles the storage of the adjoint data first in a dynamic buffer, Z, and when this buffer is filled, it dumps the buffer onto random storage file 41.

This subroutine also has an entry point SNSADJ which refines the impulse response function, $\Lambda^{\Psi_i \Omega_j}$, estimate at mid-integration interval before storing.

Description

The order of storage of information at each adjoint-solution time point in the Z buffer is listed below.

1. The adjoint variable $\lambda^{\Psi_i \Omega_j}$ matrix is stored. Each column in this matrix corresponds to a different constraint, Ψ_i; each row corresponds to an element in the state vector.
2. The impulse response function vectors are stored. These vectors, XLAMA and XLAMP contain an element for each element in the $d\Psi_i$ vector.
3. The upper triangular portion of the A matrix is stored. Only this part is needed since A is a symmetric matrix.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
A	A	I	Control integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BST03	I	A
						MTX3A	I	A
						PAY02	I	A
						SDINP	I	A
						TRAN3	I	A
IBLK1		M	Storage retrieval buffer counter	/XCODES/(	173)	AST3	M	IBLK1
						BST03	M	IBLK1
IBUFB		M	Counts number of buffers of adjoint data that have either been stored or retrieved as solution progress.	/RETREV/(	20)	BGET3	M	IBUFB
						BST03	M	IBUFB
IFB		I	File where adjoint solution is stored	/XCODES/(	140)	BGET3	I	IFB
						BST03	I	IFB
						TOPM	O	IFB
MAXB		M	Number of words in last stored partial buffer of adjoint data Corresponds to random file 41.	/RETREV/(	5)	BGET3	I	MAXB
						BST03	M	MAXB
MIXB		I	Maximum number of words in adjoint data buffer = 3000.	/RETREV/(	13)	BGET3	I	MIXB
						BST03	I	MIXB
						SDINP	I	MIXB
						TOPM	O	MIXB
MXB		I	Index of last stored word in full buffer of adjoint data.	/RETREV/(	15)	BGET3	I	MXB
						BST03	I	MXB
						SDINP	O	MXB
NBFB		I	Maximum number of buffers permitted to store adjoint solution data = 60.	/RETREV/(	11)	BST03	I	NBFB
						TOPM	O	NBFB
NBUFB		O	Number of buffers of adjoint data stored on last adjoint solution.	/RETREV/(	19)	BGET3	I	NBUFB
						BST03	O	NBUFB
NCN		I	Number of elements in dΨ	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BST03	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAY02	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BST03	I	NCNST
						CON3	I	NCNST
						PAY02	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BSTO3	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						VREF3	I	NEQ
XL	λ^{Ψ,Ω_J}	I	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
XLAA		W	Temporary storage used for estimate of midpoint impulse response function	/BSTO3 /(*	)	BSTO3	W	XLAA
XLAMA	Λ^{Ψ,Ω_J}	M	Impulse response function column vector associated with angle of attack	/AECO3 /(	16)	ADEQ3A	M	XLAMA
						ADIC3A	O	XLAMA
						AST3	O	XLAMA
						BGET3	O	XLAMA
						BSTO3	M	XLAMA
						MTX3A	I	XLAMA
						TRAN3	M	XLAMA
XLAMP	Λ^{Ψ,Ω_J}	M	Impulse response function column vector associated with bank angle	/AECO3 /(	25)	ADEQ3A	M	XLAMP
						ADIC3A	O	XLAMP
						AST3	O	XLAMP
						BGET3	O	XLAMP
						BSTO3	M	XLAMP
						MTX3A	I	XLAMP
						TRAN3	M	XLAMP
XLAP		W	Temporary storage used for estimate of midpoint impulse response function	/BSTO3 /(*	)	BSTO3	W	XLAP
Z		W	Adjoint storage buffer	/BSTO3 /(*	)	BSTO3	W	Z

```
      SUBROUTINE BST03                                                    BST03
C          STORE  ADJOINT DATA AT EACH TIME POINT                         COMN
C                                                                         COMN
      DIMENSION XLAA(9), XLAP(9)                                          BST03
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                             RETREV
      COMMON/RETREV/                                                      RETREV
     *NBUFA(2) ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA      ,  RETREV
     *MIXB     ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB        RETREV
     *,NBUFB   ,IBUFB                                                    RETREV
      COMMON/STATE3/                                                      STATE3D
     *VAR(14)  ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)   ,  STATE3D
     *XL(9,9)  ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) ,  STATE3D
     *SINPSI   ,COSPSI    ,SINRHO    ,COSRHO    ,DCORHO    ,DCORO2    ,  STATE3D
     *SVBV  (9),OMEGA     ,OMEGA2    ,                                   STATE3D
     *VDV      ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       ,  STATE3D
     *UDV      ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       ,  STATE3D
     *UDG      ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       ,  STATE3D
     *UDR      ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       ,  STATE3D
     *GDP      ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       ,  STATE3D
     *PDO      ,UDO       ,HTDV      ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                 STATE3D
      COMMON/STATE3/                                                      STATE3D
     *SIN2RO   ,COS2RO    ,COS2GM                                        STATE3D
      COMMON /XCODES/                                                     XCODES
     *ITQ   (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2),  XCODES
     *JST  (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB     ,  XCODES
     *I2OP     ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       ,  XCODES
     *IOPEN    ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    ,  XCODES
     *ITCT     ,ITER      ,IVAR      ,JK        ,JPS       ,JS        ,  XCODES
     *KOP      ,KPST      ,K         ,KST       ,NAD       ,NCASE     ,  XCODES
     *NCN      ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         ,  XCODES
     *NST      ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     ,  XCODES
     *IPHNB    ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         ,  XCODES
     *IFOB     ,NB        ,LB        ,MB        ,NPHP      ,NPHB      ,  XCODES
     *NCTIN    ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/AECO3/                                                       AECO3
     *APHO     ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       ,  AECO3
     *SINA     ,COSA      ,PHID      ,PHID      ,PHI       ,SINPHI    ,  AECO3
     *COSPHI   ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       ,  AECO3
     *CDOM     ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      ,  AECO3
     *CM       ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       ,  AECO3
     *CLAM     ,CL        ,CLA       ,CLM       ,                        AECO3
     *CD       ,CDA       ,CDM                                           AECO3
      COMMON/GENF/                                                        GENF
     *OMG(20)  ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   ,  GENF
     *A(9,9)   ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       ,  GENF
     *DTS      ,DT        ,G         ,DPSQ      ,Q         ,QS        ,  GENF
     *R        ,RE        ,MACH      ,PA        ,RO        ,CS        ,  GENF
     *VNU      ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     ,  GENF
     *SVSQ     ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     ,  GENF
     *TST(20)  ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         ,  GENF
     *TLP1(20) ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       ,  GENF
     *TIMPR    ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,  GENF
     *AE       ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     ,  GENF
     *QR       ,QV        ,FVAC      ,LIFTV     ,                        GENF
     *LIFTR    ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     ,  GENF
     *           LIFTM    ,DBR       ,DB        ,ISP       ,ISPF      ,  GENF
     *                     ULFT      ,ULFTV     ,ULFTR     ,ULFTA     ,  GENF
     *XMCG     ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,  GENF
     *CULFT    ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       ,  GENF
     *COD      ,SIDAE     ,XCG       ,ZCG       ,XJ                      GENF
      COMMON / GENF  /                                                    GENF
     *XJV      ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       ,  GENF
     *FRATED   ,IRATED    ,                                              GENF
     *P1       ,P2        ,P3       ,XK1        ,XK2       ,XK3       ,  GENF
     *XK1T     ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,  GENF
     *XK1A     ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,  GENF
     *XK1G     ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,  GENF
     *XK1R     ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      ,  GENF
     *XK1U     ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,  GENF
     *PV       ,PG        ,PP        ,PR        ,PD        ,OPDY(3,8)    GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                 MACH, MACHR,   GENF
     * ISP, ISPF, MACHV,LIFTV ,  IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                         GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                  GENF
```

```
      DIMENSION  Z(3000)                                                BST03
C                                                                       COMN
C          I    INITIALIZE  COUNTERS                                    COMN
      ENTRY BEGNB                                                       BST03
      KK= NCNST+1                                                       BST03
      IBLK1=0                                                           BST03
      IBUFB =0                                                          BST03
      RETURN                                                            BST03
C                                                                       COMN
C          II   TERMINATE  STORAGE  IF PARTIAL BUFFER, DUMP BUFFER      COMN
      ENTRY ENDB                                                        BST03
      MAXB = IBLK1                                                      BST03
       CALL IPR(2HXL,XL,I,9*NCN,0)                                      BST03
      IF(MAXB.NE.0)GO TO 10                                             BST03
      MAXB=MXB                                                          BST03
      NBUFB= IBUFB                                                      BST03
      RETURN                                                            BST03
   10 NBUFB = IBUFB +1                                                  BST03
      GO TO  90                                                         BST03
C                                                                       COMN
C          III   STORE  DATA  IN  BUFFER                                COMN
      ENTRY  BSTO                                                       BST03
   20 CONTINUE                                                          BST03
      DO 40  I=1,KK                                                     BST03
      DO 30  J=1,NEQ                                                    BST03
      IBLK1 =IBLK1+1                                                    BST03
   30 Z(IBLK1)= XL(J,I)                                                 BST03
   40 CONTINUE                                                          BST03
      DO 50  I=1,KK                                                     BST03
      IBLK1=IBLK1+1                                                     BST03
   50 Z(IBLK1) = XLAMA(I)                                               BST03
      DO 60 I=1,KK                                                      BST03
      IBLK1 =IBLK1 +1                                                   BST03
   60 Z(IBLK1)= XLAMP(I)                                                BST03
      DO 80  JJ=1,KK                                                    BST03
      DO 70  KI=1,JJ                                                    BST03
      IBLK1= IBLK1+1                                                    BST03
   70 Z(IBLK1)= A(KI,JJ)                                                BST03
   80 CONTINUE                                                          BST03
C                                                                       COMN
C          IV    IF BUFFER  FULL,DUMP BUFFER ,ELSE RETURN               COMN
      IF(IBLK1.GE.MXB) GO TO 90                                         BST03
      RETURN                                                            BST03
   90 IBUFB =IBUFB +1                                                   BST03
      IF(IBUFB.GT NBFB ) CALL STPIT(IFB)                                BST03
      CALL WRITMS(IFB,Z,MIXB,IBUFB)                                     BST03
      IBLK1=0                                                           BST03
      RETURN                                                            BST03
C                                                                       COMN
C          V    SPECIAL CODE  TO REFINE  MIDPOINT  IMPULSE-RESPONSE     COMN
C            FUNCTION    AND STORE    DATA                              COMN-
      ENTRY SNSADJ                                                      BST03
      DO 100 I=1,NCN                                                    BST03
      XLAA(I)=XLAMA(I)                                                  BST03
  100 XLAP(I)=XLAMP(I)                                                  BST03
      CALL ADEQ                                                         BST03
      DO 110 I=1,NCN                                                    BST03
      XLAMA(I)=(XLAMA(I)+XLAA(I))*.5                                    BST03
  110 XLAMP(I)=(XLAMP(I)+XLAP(I))*.5                                    BST03
      GO TO 20                                                          BST03
      END                                                               BST03
```

SUBRØUTINE
CØN3

Subroutine CON3

Purpose

Subroutine CON3 computes the constraint misses and payoff. It has two entry points the first, CONIN, sets up and computes constraint misses, ACON, for the first nominal trajectory; whereas the entry CON computes trial constraint misses and payoff, BCØN. These trial constraint misses later become nominal values in subroutine TEST if the trial trajectory is acceptable.

Description

The main logic in this subroutine is governed first by the argument list (MM, MN) which indicates partitioning of the $d\psi_i$ vector and second by the constraint variable code pointer ITQ.

MM is the first element in $d\psi_i$ to be evaluated and NN is the last.

The ITQ pointers tell whether the constraint or payoff is elapsed timed, an element of the state vector or a non-linear constraint.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ACON		M	Vector of nominal constraint misses + PAYOFF IMPROVEMENT	/GENF /(	190)	CON3	M	ACON
						TEST	M	ACON
BCON		O	Vector of constraint misses on trial trajectory	/GENF /(	199)	CON3	O	BCON
						TEST	I	BCON
						TOPM	I	BCON
DCON	$d\Psi_i$	O	Asked for correction in constraint misses and payoff vector	/GENF /(	289)	CON3	O	DCON
						MTX3A	I	DCON
						PAY02	M	DCON
						TEST	M	DCON
						TOPM	I	DCON
						TRTOSZ	I	DCON
FPD		I	Rate of change of non-linear cut-off function	/GENF /(	523)	CON3	I	FPD
						DTF3	I	FPD
						STP3	I	FPD
						VREF3	I	FPD
IOP		W	Payoff code	/CON3 /(*	)	CON3	W	IOP
ITQ		I	Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
						ADIC3A	I	ITQ
						ADID3A	I	ITQ
						CON3	I	ITQ
						SDINP	M	ITQ
						STAU	I	ITQ
						TOPM	D	IITQ
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BST03	I	NCNST
						CON3	I	NCNST
						PAY02	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST
TIME	t	I	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						VREF3	M	TIME

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
VAR	v	I	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	V
							STP3	I	VAR
							TOPM	D	KWDW
							YREF3	M	V
VARQ	(VARQ)	I	Desired constraint values [sd]		/GENF /(	61)	CON3	I	VARQ
							SDINP	M	VARQ
							TEST	I	VARQ

30 OCT 72 6 01-46

```
      SUBROUTINE CON3(MM,NN)                                              CON3
C                                                                         COMN
C       COMPUTE CONSTRAINT AND PAYOFF VALUES AT APPROPRIATE               COMN
C       ARC END  POINTS                                                   COMN
C                                                                         COMN
C       MM  IS  THE  SET NO  OF THE  FIRST CONSTRAINT                     COMN
C       NN  IS  THE  SET NO. OF THE  LAST  CONSTRAINT                     COMN
C                                                                         COMN
      COMMON /XCODES/                                                     XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGIO(20,2),JPH (20,2),  XCODES
     *JST   (20)         ,NCNST       ,NSB       ,NSAB      ,NICNB     ,  XCODES
     *I20P       ,ICOP   ,IFAW        ,IFAR      ,IFB       ,IND       ,  XCODES
     *IOPEN      ,IPH    ,ISPH        ,ISST      ,IARC      ,ISTART    ,  XCODES
     *ITCT       ,ITER   ,IVAR        ,JK        ,JPS       ,JS        ,  XCODES
     *KOP        ,KPST   ,K           ,KST       ,NAD       ,NCASE     ,  XCODES
     *NCN        ,NEQB   ,NEQ         ,NOP       ,NPH       ,N         ,  XCODES
     *NST        ,IPST   ,IPRINT      ,ISTN      ,IPHN      ,ISTNB     ,  XCODES
     *IPHNB      ,IBLK1  ,IBLK2       ,ISTOP     ,ISTPP     ,L         ,  XCODES
     *IFOB       ,NB     ,LB          ,MB        ,NPHP      ,NPHB      ,  XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/STATE3/                                                      STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)   ,  STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) ,  STATE3D
     *SINPSI     ,COSPSI   ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    ,  STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                   STATE3D
     *VDV        ,GDV      ,RDV       ,MDV       ,PDV       ,ODV       ,  STATE3D
     *UDV        ,VDG      ,GDG       ,RDG       ,PDG       ,ODG       ,  STATE3D
     *UDG        ,VDR      ,GDR       ,MDR       ,PDR       ,ODR       ,  STATE3D
     *UDR        ,VDM      ,GDM       ,MDM       ,PDM       ,VDP       ,  STATE3D
     *GDP        ,PDP      ,ODP       ,UDP       ,VDO       ,GDO       ,  STATE3D
     *PDO        ,UDO      ,HTDV      ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                 STATE3D
      COMMON/STATE3/                                                      STATE3D
     *SIN2RO     ,COS2RO    ,COS2GA                                       STATE3D
      COMMON/GENF/                                                        GENF
     *DMG(20)    ,DMGP(20,2),VARO(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   ,  GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       ,  GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        ,  GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS        ,  GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     ,  GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     ,  GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         ,  GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP       ,  GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,  GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     ,  GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     ,  GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      ,  GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     ,  GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,  GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       ,  GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                    GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       ,  GENF
     *FRATED    ,IRATED     ,                                             GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3       ,  GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,  GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,  GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,  GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      ,  GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,  GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)    GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,        GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                    FRAT
      DIMENSION TPH1(10),TST1(10)                                         GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                  GENF
C        I                                                                COMN
C       COMPUTE  CONSTRAINT  MISSES  AND  PAYOFF  FOR                     COMN
C       FIRST  NOMINAL TRAJECTORY                                         COMN
      ENTRY CONIN                                                         CON3
      ISKP=0                                                              CON3
      IF(MM.GT.NN) GO TO 50                                               CON3   50
      DO 40 I=MM,NN                                                       CON3
      JJ = ITQ(I) -1                                                      CON3
      IF(JJ.EQ.0) GO TO 30                                                CON3   30
```

```
      IF(JJ.LE. 9) GO TO 10                                             CON3
      CALL PDBC(ITQ(I),Z,SX,FPD,0,ISKP)                                 CON3
      ACON(I) = VARQ(I) - Z                                             CON3
      GO TO  20                                                         CON3
   10 CONTINUE                                                          CON3
      IF(JJ.EQ 9) JJ=8                                                  CON3
      ACON(I)=VARQ(I)-VAR(JJ)                                           CON3
   20 DCON(I) =ACON(I)                                                  CON3
      GO TO 40                                                          CON3
   30 ACON(I) =VARQ(I)- TIME                                            CON3
      DCON(I) = ACON(I)                                                 CON3
   40 CONTINUE                                                          CON3
   50 IF(NN LT NCNST) RETURN                                            CON3
   60 IOP= ITQ(NCNST+1) -1                                              CON3
      IF(IOP.EQ.0) GO TO 80                                             CON3
      IF(IOP.LE.9) GO TO 70                                             CON3
      CALL PDBC( ITQ(NCNST+1),Z,SX,FPD,0,ISKP)                          CON3
      ACON(NCNST+1)= Z                                                  CON3
      RETURN                                                            CON3
   70 CONTINUE                                                          CON3
        IF(IOP.EQ.9) IOP=8                                              CON3
      ACON(NCNST+1) = VAR(IOP)                                          CON3
      RETURN                                                            CON3
   80 ACON(NCNST+1) = TIME                                              CON3
      RETURN                                                            CON3
C                                                                       CONN
C       II   CONSTRAINT MISSES  AND  PAYOFF  FOR TRIAL  TRAJECTORY      CONN
C                                                                       CONN
      ENTRY CON                                                         CON3
      ISKP=0                                                            CON3
      IF(MM.GT.NN) GO TO 120                                            CON3
      DO  110  I=MM,NN                                                  CON3
      JJ =  ITQ(I)-1                                                    CON3
      IF(JJ.EQ.0) GO TO 100                                             CON3
      IF(JJ.LE  9)  GO  TO  90                                          CON3
      CALL PDBC(ITQ(I),Z,SX,FPD,0,ISKP)                                 CON3
      BCON(I)= VARQ(I) - Z                                              CON3
      GO   TO 110                                                       CON3
   90 CONTINUE                                                          CON3
      IF(JJ.EQ.9) JJ=8                                                  CON3
      BCON(I) = VARQ(I)- VAR(JJ)                                        CON3
      GO TO 110                                                         CON3
  100 BCON(I) = VARQ(I)- TIME                                           CON3
  110 CONTINUE                                                          CON3
  120 IF(NN LT.NCNST) RETURN                                            CON3
  130 IOP = ITQ(NCNST+1)                                                CON3
      IF(IOP.EQ.1) GO TO 150                                            CON3
      IF (IOP LE.10)  GO TO 140                                         CON3
      CALL PDBC( ITQ(NCNST+1),Z,SX,FPD,0,ISKP)                          CON3
      BCON(NCNST+1)= Z                                                  CON3
      RETURN                                                            CON3
  140 CONTINUE                                                          CON3
      IF(IOP.EQ 10) IOP=9                                               CON3
      BCON(NCNST +1) = VAR(IOP-1)                                       CON3
      RETURN                                                            CON3
  150 BCON(NCNST +1) = TIME                                             CON3
      RETURN                                                            CON3
      END                                                               CON3
```

20 OCT 72 G 01-46

SUBRØUTINE
DER3A

Subroutine DER3A

Entry Points DER

Purpose

Subroutine DER3A computes the derivatives of the equations of motion for the forward steepest descent trajectory.

Description

The equations programmed in subroutine DER3A are given in Volume I, Section 2. 3. The acceleration vector component AV, AG, AP and AM are computed in subroutine ACCEL.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AG	a^{γ}	I	Acceleration vector element.		/AXL /(	2)	ACCEL	M	AG
							DER3A	I	AG
							PDY3A	I	AG
AM	a^{m}	I	Acceleration vector element.		/AXL /(	4)	ACCEL	M	AM
							DER3A	I	AM
AP	a^{ψ}	I	Acceleration vector element.		/AXL /(	3)	ACCEL	M	AP
							DER3A	I	AP
							PDY3A	I	AP
AV	a^{v}	I	Acceleration vector element.		/AXL /(	1)	ACCEL	M	AV
							DER3A	I	AV
COSGAM	$\cos(\gamma)$	I	See symbol		/STATE3/(	687)	ACCEL	I	COSGAM
							BL4	I	COSGAM
							BL8	I	COSGAM
							DER3A	I	COSGAM
							EQUA3	O	COSGAM
							MODELA	I	COSGAM
							MODELB	I	COSGAM
							OUT	I	COSGAM
							PDBC	I	COSGAM
							PDY3A	I	COSGAM
COSPSI	$\cos(\psi)$	I	See symbol		/STATE3/(	705)	BL4	I	COSPSI
							BL7	I	COSPSI
							BL8	I	COSPSI
							DER3A	I	COSPSI
							EQUA3	O	COSPSI
							MODELA	I	COSPSI
							MODELB	I	COSPSI
							PDBC	I	COSPSI
							PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol		/STATE3/(	707)	BL4	I	COSRHO
							BL7	I	COSRHO
							BL8	I	COSRHO
							DER3A	I	COSRHO
							EQUA3	M	COSRHO
							MODELA	I	COSRHO
							MODELB	I	COSRHO
							OUT	I	COSRHO
							PDBC	I	COSRHO
							PDY3A	I	COSRHO
G	g	I	Gravitational attraction	(FT/SEC**2)	/GENF /(	301)	BL4	I	G
							BL7	I	G
							BL8	I	G
							DER3A	I	G
							EQUA3	M	G
							MODELA	I	G
							MODELB	I	G
							PDY3A	I	G
							SDER3	I	G
							SDINP	M	G
GD	$\dot{\gamma}$	O	GAM derivative		/STATE3/(	16)	DER3A	O	GD
							PDBC	I	GD
GMDOT	$\dot{\gamma}^{*}$	I	Pitch rate	(DEG/SEC)	/ARCDAT/(	15)	DER3A	I	GMDOT
							MODELA	I	GMDOT
							MODELB	I	GMDOT
							PROPB	I	GMDOT
							PROPIN	I	GMDOT
HD	$\dot{h}$	O	ALT derivative		/STATE3/(	17)	DER3A	O	HD
							PDBC	I	HD
HTD	$\dot{Q}$	O	Heating derivative		/STATE3/(	22)	DER3A	O	HTD
							OUT	I	HTD
							PDBC	I	HTD
							PDY3A	M	HTD

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
JGII		I	Control option	/XCODES/(	195)	ACCEL	I	JGII
						BNTG	O	JGII
						DER3A	I	JGII
						FNTG	M	JGII
						GUI3A	I	JGII
						MODELA	I	JGII
						MODELB	I	JGII
						MTX3A	I	JGII
						PDY3A	I	JGII
JPRP		I	Propulsion flag for different rocket options	/XCODES/(	194)	ACCEL	I	JPRP
						DER3A	I	JPRP
						EQUA3	I	JPRP
						MODELA	I	JPRP
						PDY3A	I	JPRP
						PROPB	O	JPRP
						PROPIN	O	JPRP
MD	$\dot{m}$	O	MASS derivative	/STATE3/(	18)	DER3A	O	MD
						OUT	I	MD
OCORHO	$\omega \times COSRHO$	I	See symbol	/STATE3/(	708)	DER3A	I	OCORHO
						EQUA3	M	OCORHO
						MODELA	I	OCORHO
						MODELB	I	OCORHO
						PDBC	I	OCORHO
						PDY3A	I	OCORHO
OCORO2	$\omega \times OCORHO$	I	See symbol	/STATE3/(	709)	DER3A	I	OCORO2
						EQUA3	O	OCORO2
						MODELA	I	OCORO2
						MODELB	I	OCORO2
						PDY3A	I	OCORO2
OD	$\dot{\rho}$	O	Latitude derivative	/STATE3/(	20)	DER3A	O	OD
OMGZ	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
						CRASH	I	OMEGA
						DER3A	I	OMGZ
						EQUA3	I	OMGZ
						GEINP	I	OMGZ
						MODELA	I	OMGZ
						MODELB	I	OMGZ
						PDBC	I	OMGZ
						PDY3A	I	OMGZ
						SDINP	I	OMGZ
						TOPM	I	OMGZ
PSID	$\dot{\psi}$	O	Azimuth derivative	/STATE3/(	19)	DER3A	O	PSID
QMULT	=0 OR 1	I	Heating flag multiplier	/ARCDAT/(	40)	DER3A	I	QMULT
						FXDAT	O	QMULT
						PDY3A	I	QMULT
R	R	I	Radial distance from earth center to vehicle (FT)	/GENF /(	305)	BL4	I	R
						BL7	I	R
						BL8	I	R
						DER3A	I	R
						EQUA3	M	R
						MODELA	I	R
						MODELB	I	R
						PDBC	I	R
						PDY3A	I	R
						TRTOSZ	I	R
RDI		I	Angle to radian conversion, 01745329252	/DATA /(	3)	BLICO	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	D	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
RHOB	ρ_b	I	Atmosphere base density for heating calculation (LB/FT**3)	/ARCDAT/(	39)	DER3A	I	RHOB
						FXDAT	O	RHOB
						PDY3A	I	RHOB
RO	ρ_a	I	Atmospheric density (SLUGS/FT**3)	/GENF /(	309)	BL7	I	RO
						BL8	I	RO
						DER3A	I	RO
						EQUA3	I	RO
						OUT	I	RO
						PDBC	I	RO
						PDY3A	I	RO
SINGAM	$\sin(\gamma)$	I	See symbol	/STATE3/(	688)	BL4	I	SINGAM
						BL7	I	SINGAM
						BL8	I	SINGAM
						DER3A	I	SINGAM
						EQUA3	O	SINGAM
						MODELA	I	SINGAM
						MODELB	I	SINGAM
						PDBC	I	SINGAM
						PDY3A	I	SINGAM
						SDER3	I	SINGAM
SINPSI	$\sin(\psi)$	I	See symbol	/STATE3/(	704)	BL4	I	SINPSI
						BL7	I	SINPSI
						BL8	I	SINPSI
						DER3A	I	SINPSI
						EQUA3	O	SINPSI
						MODELA	I	SINPSI
						MODELB	I	SINPSI
						PDBC	I	SINPSI
						PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol	/STATE3/(	706)	BL4	I	SINRHO
						BL7	I	SINRHO
						BL8	I	SINRHO
						DER3A	I	SINRHO
						EQUA3	O	SINRHO
						MODELA	I	SINRHO
						MODELB	I	SINRHO
						OUT	I	SINRHO
						PDBC	I	SINRHO
						PDY3A	I	SINRHO
SQRT		F	Square root function	/SQRT /(5	)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTOE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAYO2	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
UD	$\dot{\mu}$	O	Longitude derivative	/STATE3/(	21)	DER3A	O	UD
						PDBC	I	UD

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE		SUBROUTINE USAGE		
				BLOCK	LOC	SUBR	CODE	VAR
V	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V
VD	$\dot{y}$	O	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT

```
      SUBROUTINE DER3A                                                   DER3A
C                                                                        COMN
C       COMPUTE DERIVATIVES OF EQUATIONS OF MOTION                       COMN
C                                                                        COMN
      COMMON/DATA/                                                       DATA
     *PI        ,RAD        ,RDI        ,SC         ,UMF        ,IMPF      ,  DATA
     *FTNM      ,CAR        ,JOP1       ,JOP2       ,JOP3       ,JOP4        DATA
      COMMON /XCODES/                                                    XCODES
     *ITQ    (9),ICOR   (20),ITI        ,INTB       ,JGID(20,2),JPH (20,2),  XCODES
     *JST   (20)            ,NCHST      ,NSB        ,NSAB       ,NICNB     ,  XCODES
     *I2OP      ,ICOP       ,IFAW       ,IFAR       ,IFB        ,IND       ,  XCODES
     *IOPEN     ,IPH        ,ISPH       ,ISST       ,IARC       ,ISTART    ,  XCODES
     *ITCT      ,ITER       ,IVAR       ,JK         ,JPS        ,JS        ,  XCODES
     *KOP       ,KPST       ,K          ,KST        ,NAD        ,NCASE     ,  XCODES
     *NCN       ,NEQB       ,NEQ        ,NOP        ,NPH        ,N         ,  XCODES
     *NST       ,IPST       ,IPRINT     ,ISTN       ,IPHN       ,ISTNB     ,  XCODES
     *IPHNB     ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP      ,L         ,  XCODES
     *IFOB      ,NB         ,LB         ,MB         ,NFHP       ,NPHB      ,  XCODES
     *NCTIN     ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3            XCODES
      COMMON/STATE3/                                                     STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)      ,SVY(10)   ,  STATE3D
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15) ,  STATE3D
     *SINPSI    ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2    ,  STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2     ,                                    STATE3D
     *VDV       ,GDV        ,RDV        ,MDV        ,PDV        ,ODV       ,  STATE3D
     *UDV       ,VDG        ,GDG        ,RDG        ,PDG        ,ODG       ,  STATE3D
     *UDG       ,VDR        ,GDR        ,MDR        ,PDR        ,ODR       ,  STATE3D
     *UDR       ,VDM        ,GDM        ,MDM        ,PDM        ,VDP       ,  STATE3D
     *GDP       ,PDP        ,ODP        ,UDP        ,VDO        ,GDO       ,  STATE3D
     *PDO       ,UDO        ,HTDV       ,HTDR                                STATE3D
      REAL MDM , MDV, MDR                                                STATE3D
      COMMON/STATE3/                                                     STATE3D
     *SIN2RO    ,COS2RO     ,COS2GM                                          STATE3D
      COMMON/GENF/                                                       GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)    ,TOL(9)     ,SVAR(10)   ,WDC(20)   ,  GENF
     *A(9,9)    ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)    ,DTP       ,  GENF
     *DTS       ,DT         ,G          ,DPSQ       ,Q          ,QS        ,  GENF
     *R         ,RE         ,MACH       ,PA         ,RO         ,CS        ,  GENF
     *VNU       ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ     ,  GENF
     *SVSQ      ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)     ,  GENF
     *TST(20)   ,TPH    (20),DIS(20)    ,DIP(20)    ,T          ,W         ,  GENF
     *TLP1(20)  ,TLS1   (20),DIP1(20)   ,DIS1(20)   ,TIME       ,OMP       ,  GENF
     *TIMPR     ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)   ,  GENF
     *AE        ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV     ,  GENF
     *QR        ,QV         ,FVAC       ,LIFTV      ,                        GENF
     *LIFTR     ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA     ,  GENF
     *           LIFTM      ,DBR        ,DB         ,ISP        ,ISPF      ,  GENF
     *                       ULFT       ,ULFTV      ,ULFTR      ,ULFTA     ,  GENF
     *XMCG      ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE     ,  GENF
     *CULFT     ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SID       ,  GENF
     *COD       ,SIDAE      ,XCG        ,ZCG        ,XJ                      GENF
      COMMON / GENF  /                                                   GENF
     *XJV       ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP       ,  GENF
     *FRATED    ,IRATED     ,                                                GENF
     *P1        ,P2         ,P3        ,XK1         ,XK2        ,XK3       ,  GENF
     *XK1T      ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D      ,  GENF
     *XK1A      ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V      ,  GENF
     *XK1G      ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P      ,  GENF
     *XK1R      ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O      ,  GENF
     *XK1U      ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M      ,  GENF
     *PV        ,PG         ,PP         ,PR         ,PD         ,OPOY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                        GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                 GENF
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,   EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SO2),  EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),  EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SO2D)             EQUV3
      REAL M,MU,MD                                                       EQUV3
      COMMON/GLOBAL/                                                     GLOBAL
     *GR        ,ER         ,OMGZ       ,XLAMRF     ,YMURF      ,LUM         GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC        ,NBRAN     ,NFARC      ,ID(4)       GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG        ,MAXTAB                               GLOBAL
     *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)         GLOBAL
```

```
      *,ITPSO ,XSOL    ,KGLOBL(8)                                              RETAP
       COMMON/ARCDAT/                                                          ARCDAT
      *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      ,  ARCDAT
      *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      ,  ARCDAT
      *XLMAX     ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      ,  ARCDAT
      *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      ,  ARCDAT
      *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      ,  ARCDAT
      *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        ,  ARCDAT
      *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                    ARCDAT
      * ,FRATE  ,  ARCD(9)                                                     RETAP
       DIMENSION  ARCDA(40)                                                    ARCDAT
       EQUIVALENCE(SREF,ARCDA)                                                 ARCDAT
       COMMON/AECO3/                                                           AECO3
      *APHO      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       ,  AECO3
      *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI    ,  AECO3
      *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       ,  AECO3
      *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      ,  AECO3
      *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       ,  AECO3
      *CLAM      ,CL        ,CLA       ,CLM       ,                          AECO3
      *CD        ,CDA       ,CDM                                              AECO3
       COMMON/ AXL/                                                            AXL
      *AV   ,AG   ,AP   ,AM   ,AVV  ,AGV  ,APV  ,AMV  ,AVG  ,AGG  ,APG  ,  AXL
      *AMG  ,AVP  ,AGP  ,APP  ,AMP  ,AVR  ,AGR  ,APR  ,AMR  ,AVO  ,AGO  ,  AXL
      *APO  ,AMO  ,AVU  ,AGU  ,APU  ,AMU  ,AVM  ,AGM  ,APM  ,AMM            AXL
       DIMENSION  AY(1)                                                        AXL
       EQUIVALENCE (AY,AVV)                                                    AXL
       EQUIVALENCE (PSID,PD)                                                   IO
       ENTRY DER                                                               DER3A
C    I   IF  ROCKET COMPUTE  SPECIFIC IMPULSE                                  COMN
       IF(JPRP.GT 1) CALL  IMPUL                                               COMN
C                                                                              COMN
C       II   COMPUTE  ACCELERATION  VECTOR                                     COMN
       CALL  ACCEL                                                             IO
       VD=  R*OCORO2*( COSRHO*SINGAM- SINRHO*COSPSI*COSGAM) -G*SINGAM+         DER3A
      1    AV                                                                  DER3A
C                                                                              COMN
C       III    VERT  RISE /PITCHOVER  DERIVATIVES                              COMN
       IF(JGII          .NE.8) GO TO 10                                        DER3A
       GD = GMDOT*RDI                                                          DER3A
       PSID = 0.                                                               DER3A
       GO TO 20                                                                DER3A
    10 CONTINUE                                                                DER3A
       GD=  OCORHO*(2 *SINPSI+R*OMGZ/V*(COSRHO*COSGAM+ SINRHO*COSPSI*SING DER3A
      1AM)) + COSGAM*(V/R-G/V) + AG                                            DER3A
      *  / V                                                                   IO
       PSID = OCORHO*( R* OMGZ*SINRHO*SINPSI/V-2 *COSPSI*SINGAM)/ COSGAM+ DER3A
      1 SINRHO*(V*COSGAM*SINPSI/(R*COSRHO)+2*OMGZ) + AP                        DER3A
      *  / ( V*COSGAM)                                                         IO
    20 CONTINUE                                                                DER3A
       RD= AM                                                                  DER3A
       HD = V*SINGAM                                                           DER3A
       OD= V* COSGAM*COSPSI/R                                                  DER3A
       UD= V*COSGAM*SINPSI/(R*COSRHO)                                          DER3A
       IF(QMULT*RO EQ 0 ) RETURN                                               APR27
       HTD= QMULT*  17600 *SQRT(RO/RHOB) *(V/26000.)**3 15                     DER3A
       RETURN                                                                  DER3A
       END                                                                     DER3A
```

SUBRØUTINE DTF3

Subroutine DTF3

Entry Point. DTF

Purpose

Subroutine DTF3 computes integration interval to approximately hit a state or function cut-off.

Description

After the forward integration controlling subprogram, FNTG, detects that an arc cut-off is imminent, DTF3 is called to estimate the integration interval to satisfy the cut-off function. This is done using the equations described in Volume I, Section 15.2, equations 15.2-3 through 15.2-11.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
AA		W	Coefficients in compute interval formula	/DTF3	/(*)	DTF3	W	AA
A2		W	Coefficients in compute interval formula	/DTF3	/(*)	DTF3	W	A2
B		W	Coefficients in compute interval formula	/DTF3	/(*)	DTF3	W	B
C		W	Coefficients in compute interval formula	/DTF3	/(*)	DTF3	W	C
FP		I	Current value of cut-off function - non-linear only	/GENF	/(521)	DTF3	I	FP
						STP3	I	FP
						TOL3	I	FP
						YREF3	I	FP
FPD		I	Rate of change of non-linear cut-off function	/GENF	/(523)	CON3	I	FPD
						DTF3	I	FPD
						STP3	I	FPD
						YREF3	I	FPD
FPOLD		I	Value of non-linear cut-off function at prior compute interval	/GENF	/(522)	DTF3	I	FPOLD
						STP3	O	FPOLD
						TOL3	O	FPOLD
						YREF3	I	FPOLD
H		W	Change in Ω from last to current compute interval	/DTF3	/(*)	DTF3	W	H
HP		W	Current error in Ω	/DTF3	/(*)	DTF3	W	HP
REZ		W	Ratio of HP/H	/DTF3	/(*)	DTF3	W	REZ
SVY	y_{-1}	I	State and time array at previous compute interval	/STATE3	/(236)	DTF3	I	SVY
						REU3	M	SVY
						YREF3	I	SVY
TIME	t	I	Time (elapsed)	/GENF	/(493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
V	v	I	Relative velocity (FT/SEC)	/STATE3	/(1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
VT	$\dot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT

```
      FUNCTION DTF3( II,OM)                                                DTF3
C       COMPUTES  INTEGRATION INTERVAL  TO  APPROXIMATELY                  COMN
C       HIT  STATE OR FUNCTION  CUT OFF                                    COMN
      COMMON/STATE3/                                                       STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)      ,SVY(10)   , STATE3D
     *XL(9,9)    ,YOP(20,9),YDS (20,9),COSGAM    ,SINGAM     ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO     ,OCORO2    , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2    ,                                   STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV        ,ODV       , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG        ,ODG       , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR        ,ODR       , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM        ,VDP       , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO        ,GDO       , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                  STATE3D
      COMMON/STATE3/                                                       STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                        STATE3D
      COMMON/GENF/                                                         GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)   ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)    ,DTP       , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q          ,QS        , GENF
     *R          ,RE        ,MACH      ,PA        ,RO         ,CS        , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR        ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS        ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T          ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME       ,OMP       , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN      ,TBU(20)   , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR      ,MACHV     , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                        GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR      ,DRAGA     , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP        ,ISPF      , GENF
     *                       ULFT      ,ULFTV     ,ULFTR      ,ULFTA     , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM      ,CODAE     , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE     ,SID       , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                      GENF
      COMMON / GENF  /                                                     GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG        ,XKP       , GENF
     *FRATED   ,IRATED      ,                                              GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2        ,XK3       , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D       ,XK3D      , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V       ,XK3V      , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P       ,XK3P      , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O       ,XK3O      , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M       ,XK3M      , GENF
     *PV         ,PG        ,PP        ,PR        ,PO         ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                     FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      EQUIVALENCE (VAR,V),(DVAR,VT)                                        DTF3
      DIMENSION  V(4), VT(4)                                               DTF3
      ENTRY DTF                                                            DTF3
      I=II                                                                 DTF3
C                                                                          COMN
C     I  TEST  FOR  FUNCTION  CUT-OFF  YES  GO TO   III                    COMN
      IF(I.GT.7) GO TO  20                                                 DTF3   20
C                                                                          COMN
C       II     STATE   CUT OFF   PRELIM  CALC.                             COMN
      H= V(I) - SVY(I+1)                                                   DTF3
      HP = OM  - V(I)                                                      DTF3
      VTI = VT(I)                                                          DTF3
   10 CONTINUE                                                             DTF3
C                                                                          COMN
C       II-A    TEST  FOR  PASSING  CUT-OFF                                COMN
      IF ( ABS(H).LT.1.E-10) GO TO 30                                      DTF3   30
C                                                                          COMN
C       II-B     CALCULATE  TRUNCATED  INTEGRATION  STEP                   COMN
C                USING  QUADRATIC  APPROXIMATION                           COMN
      REZ= HP/H                                                            DTF3
      AA= (1.+REZ)                                                         DTF3
      A2=REZ*REZ                                                           DTF3
      C =  A2*AA                                                           DTF3
      B =AA*(1.-A2)                                                        DTF3
      AA= 1.-B                                                             DTF3
      DTF3= AA*SVY(1)+B*TIME +C*H /VTI - TIME                              DTF3
```

```
      RETURN                                                            DTF3
C                                                                       COMN
C       III   FUNCTION  CUT OFF  PRELIM. CALC                           COMN
   20 H= FP-FPOLD                                                       DTF3
      HP=DM -FP                                                         DTF3
      VTI= FPD                                                          DTF3
      GO TO 10                                                          DTF3
   30 DTF =0.                                                           DTF3
      RETURN                                                            DTF3
      END                                                               DTF3
```

SUBRØUTINE EL1

Subroutine EL1

Entry EL1001, EL1000

Purpose

Subroutine EL1 is a governing equation for the computation of the engine. deflection component of the in-plane control vector. It corresponds to fixed engine deflection.

Description

The in-plane control vector computation is described in Volume I, Section 9.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
COE		I	Constant value of engine deflection	(RAD)	/GENF	/(553)	EL1	I	COE
							PROPB	O	COE
							PROPIN	O	COE
DELTAE	δ_E	I	Engine gimbal deflection angle	(RAD)	/GENF	/(554)	BLGCON	M	DELTAE
							EL1	I	DELTAE
							OUT	I	DELTAE
							REU3	O	DELTAE
							VT	I	DELTAE
XK2		O	Second control vector governing equation value. If non-trivial corresponds to error in moment balance		/GENF	/(573)	EL1	O	XK2
							EL2	O	XK2
XK2D		O	Partial of governing equation wrt state or control vector component		/GENF	/(579)	ACCEL	I	XK2D
							BLGCON	I	XK2D
							EL1	O	XK2D
							EL2	O	XK2D

```
      SUBROUTINE EL1                                                  EL1
      COMMON/GENF/                                                    GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)   ,SVAR(10)  ,WDC(20)  ,  GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)  ,DTP      ,  GENF
     *DTS        ,DT        ,G         ,DPSQ     ,Q         ,QS       ,  GENF
     *R          ,RE        ,MACH      ,PA       ,RO        ,CS       ,  GENF
     *VNU        ,PAR       ,ROR       ,CSR      ,VNR       ,SUMSQ    ,  GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP      ,TOS       ,TR(9)    ,  GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)  ,T         ,W        ,  GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20) ,TIME      ,OMP      ,  GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX      ,TBURN     ,TBU(20)  ,  GENF
     *AE         ,FP        ,FPOLD     ,FPD      ,MACHR     ,MACHV    ,  GEPF
     *QR         ,OV        ,FVAC      ,LIFTV    ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV    ,DRAGR     ,DRAGA    ,  GENF
     *            LIFTM     ,DBR       ,DB       ,ISP       ,ISPF     ,  GENF
     *                       ULFT      ,ULFTV    ,ULFTR     ,ULFTA    ,  GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA    ,XMCGM     ,CODAE    ,  GENF
     *CULFT      ,CT        ,CALPHA    ,CDE      ,DELTAE    ,SID      ,  GENF
     *COD        ,SIDAE     ,XCG       ,ZCG      ,XJ        ,            GENF
      COMMON / GENF  /                                                GENF
     *XJV        ,XJR       ,GH        ,GAMMAD   ,XKG       ,XKP      ,  GENF
     *FRATED     ,IRATED    ,                                           GENF
     *P1         ,P2        ,P3       ,XK1       ,XK2       ,XK3      ,  GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D     ,XK2D      ,XK3D     ,  GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V     ,XK2V      ,XK3V     ,  GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P     ,XK2P      ,XK3P     ,  GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O     ,XK2O      ,XK3O     ,  GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M     ,XK2M      ,XK3M     ,  GENF
     *PV         ,PG        ,PP        ,PR       ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,              MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                 FRAT
      DIMENSION TPH1(10),TST1(10)                                     GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                              GENF
      COMMON/AEC03/                                                   AEC03
     *APHO       ,APHR      ,ALPHA     ,VDA      ,GDA       ,PDA      ,  AEC03
     *SINA       ,COSA      ,PHIO      ,PHID     ,PHI       ,SINPHI   ,  AEC03
     *COSPHI     ,GDPH      ,PDPH      ,XLAMA(9) ,XLAMP(9)  ,COO      ,  AEC03
     *CDOM       ,CLO       ,FK        ,XCGM     ,ZCGM      ,CLOM     ,  AEC03
     *CM         ,CMA       ,CMAM ,CMM ,CMO      ,CMOM      ,FKM      ,  AEC03
     *CLAM       ,CL        ,CLA       ,CLM      ,                       AEC03
     *CD         ,CDA       ,CDM                                        AEC03
      ENTRY EL1001                                                    EL1
   40 XK2D = 1.                                                       EL1
      ENTRY EL1000                                                    EL1
   50 XK2 = DELTAE - CDE                                              EL1
C                                                                     EL1
      RETURN                                                          EL1
      END                                                             EL1
```

20 OCT 72 G.01-46

390

SUBRØUTINE EL2

Subroutine EL2

Entry Points. EL2010, EL2001, EL2000

Purpose

Subroutine EL2 is a governing equation for the computation of the engine deflection component of the in-plane control vector.

It corresponds to variable engine deflection to balance all or part of the aerodynamic moment.

Description

The governing equation, XK2, computed in EL2, represents the balance of the aerodynamic moment given in Volume I, equation 3. 1-12 by the engine moment as given in 3. 1-17. The complete description of the in-plane control vector calculation is given in Section 9 of Volume I. The entry points, are used for separating the calculation of partials derivatives of XK2 with respect to in-plane control vector elements and with respect to the state vector.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
COD	$\cos(\delta_E)$	I	See symbol	/GENF /(	556)	EL2	I	COD
						OUT	I	COD
						VT	M	COD
SID	$\sin(\delta_E)$	I	See symbol	/GENF /(	555)	EL2	I	SID
						OUT	·I	SID
						VT	M	SID
T	T	I	Thrust (LBS)	/GENF /(	411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SOER3	I	T
XCG	X_{CG}	I	Center of gravity body x station (FT)	/GENF /(	558)	EL2	I	XCG
						EQUA3	I	XCG
						VT	I	XCG
XCGM	$\partial X_{CG}/\partial m$	I	See symbol	/AECO3 /(	38)	EL2	I	XCGM
						EQUA3	O	XCGM
						VT	I	XCGM
XE	X_E	I	Engine thrust centroid body x station	/ARCDAT/(	34)	EL2	I	XE
XJ	J	I	Control blend factor	/GENF /(	560)	EL2	I	XJ
						EQUA3	I	XJ
						OUT	I	XJ
						VT	I	XJ
XJR		I	Partial of blend factor wrt altitude	/GENF /(	562)	EL2	I	XJR
						EQUA3	O	XJR
						VT	I	XJR
XJV		I	Partial of blend factor wrt velocity	/GENF /(	561)	EL2	I	XJV
						EQUA3	O	XJV
						VT	I	XJV
XK2		O	Second control vector governing equation value. If non-trivial corresponds to error in moment balance	/GENF /(	573)	EL1	O	XK2
						EL2	O	XK2
XK2A		O	Partial of governing equation wrt state or control vector component	/GENF /(	582)	ACCEL	I	XK2A
						BLGCON	I	XK2A
						EL2	O	XK2A
XK2D		O	Partial of governing equation wrt state or control vector component	/GENF /(	579)	ACCEL	I	XK2D
						BLGCON	I	XK2D
						EL1	O	XK2D
						EL2	O	XK2D
XK2M		O	Partial of governing equation wrt state or control vector component	/GENF /(	603)	EL2	O	XK2M
XK2R		O	Partial of governing equation wrt state or control vector component	/GENF /(	594)	EL2	O	XK2R
XK2T		O	Partial of governing equation wrt state or control vector component	/GENF /(	576)	BLGCON	I	XK2T
						EL2	O	XK2T
XK2V		O	Partial of governing equation wrt state or control vector component	/GENF /(	585)	EL2	O	XK2V
XMCG	M_{CG}	I	Aerodynamic moment about center of gravity (FT-LBS)	/GENF /(	544)	EL2	I	XMCG
						OUT	I	XMCG
						VT	M	XMCG
XMCGA		I	Partial of XMCG wrt angle of attack	/GENF /(	547)	EL2	I	XMCGA
						VT	M	XMCGA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBR	CODE	VAR
XMCGM		I	Partial of XMCG wrt mass		/GENF /(	548)	EL2 VT	I M	XMCGM XMCGM
XMCGR		I	Partial of XMCG wrt altitude		/GENF /(	546)	EL2 VT	I M	XMCGR XMCGR
XMCGV		I	Partial of XMCG wrt velocity		/GENF /(	545)	EL2 VT	I M	XMCGV XMCGV
ZCG	Z_{CG}	I	Center of gravity body z station	(FT)	/GENF /(	559)	EL2 EQUA3 VT	I I I	ZCG ZCG ZCG
ZCGM	$\partial Z_{CG}/\partial m$	I	See symbol		/AECO3 /(	39)	EL2 EQUA3 VT	I O I	ZCGM ZCGM ZCGM
ZE	Z_E	I	Engine thrust centroid body z station		/ARCDAT/(	35)	EL2 VT	I I	ZE ZE

```
      SUBROUTINE EL2                                                    EL2
      COMMON/GENF/                                                      GENF
     *OMG(20)     ,OMGP(20,2),VARQ(9)    ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)      ,ACON(9)   ,BCON(9)    ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS         ,DT        ,G          ,DPSQ      ,Q         ,QS        , GENF
     *R           ,RE        ,MACH       ,PA        ,RO        ,CS        , GENF
     *VNU         ,PAR       ,ROR        ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ        ,TIMEPH    ,TIMES      ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)     ,TPH   (20),DIS(20)    ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)    ,TLS1  (20),DIF1(20)   ,DIS1(20)  ,TIME      ,OMP       , GENF
     *TIMPR       ,LIFT      ,DRAG       ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE          ,FP        ,FPOLD      ,FPD       ,MACHR     ,MACHV     , GENF
     *QR          ,QV        ,FVAC       ,LIFTV     ,                       GENF
     *LIFTR       ,LIFTA     ,            DRAGV     ,DRAGR     ,DRAGA     , GENF
     *             LIFTM     ,DBR        ,DB        ,ISP       ,ISPF      , GENF
     *                        ULFT       ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG        ,XMCGV     ,XMCGR      ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT       ,CT        ,CALPHA     ,CDE       ,DELTAE    ,SID       , GENF
     *COD         ,SIDAE     ,XCG        ,ZCG       ,XJ                     GENF
      COMMON / GENF /                                                   GENF
     *XJV         ,XJR       ,GH         ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED      ,IRATED    ,                                              GENF
     *P1          ,P2        ,P3        ,XK1        ,XK2       ,XK3       , GENF
     *XK1T        ,XK2T      ,XK3T       ,XX1D      ,XK2D      ,XK3D      , GENF
     *XK1A        ,XK2A      ,XK3A       ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G        ,XK2G      ,XK3G       ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R        ,XK2R      ,XK3R       ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U        ,XK2U      ,XK3U       ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV          ,PG        ,PP         ,PR        ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                      FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)     ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)      ,SVY(10)   , STATE3D
     *XL(9,9)     ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI      ,COSPSI    ,SINRHO     ,COSRHO    ,OCORHO    ,OCORO2    , STATE3D
     *SVBV    (9),OMEGA      ,OMEGA2     ,                                  STATE3D
     *VDV         ,GDV       ,RDV        ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV         ,VDG       ,GOG        ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG         ,VDR       ,GDR        ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR         ,VDM       ,GDM        ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP         ,PDP       ,ODP        ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO         ,UDO       ,HTDV       ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO      ,COS2RO    ,COS2GA                                        STATE3D
      COMMON/AECO3/                                                     AECO3
     *APHO        ,APHR      ,ALPHA      ,VDA       ,GDA       ,PDA       , AECO3
     *SINA        ,COSA      ,PHIO       ,PHID      ,PHI       ,SINPHI    , AECO3
     *COSPHI      ,GDPH      ,PDPH       ,XLAMA(9)  ,XLAMP(9)  ,CDO       , AECO3
     *CDOM        ,CLO       ,FK         ,XCGM      ,ZCGM      ,CLOM      , AECO3
     *CM          ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       , AECO3
     *CLAM        ,CL        ,CLA        ,CLM       ,                       AECO3
     *CO          ,CDA       ,CDM                                           AECO3
C                                                                       EL2
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF        ,EJ        ,XISP       ,TMULT     ,DTNC      ,DTPI      , ARCDAT
     *IATM        ,IMODE     ,JAER       ,JPRO      ,QMAX      ,GMAX      , ARCDAT
     *XLMAX       ,HDMAX     ,GMDOT      ,ALFMAX    ,PHMAX     ,MAEA      , ARCDAT
     *MAEB        ,MAEC      ,MAED       ,MAEE      ,MAEF      ,MAEG      , ARCDAT
     *MT          ,MISP      ,MXCG       ,MZCG      ,MWDA      ,MWDB      , ARCDAT
     *MDB         ,XCGR      ,ZCGR       ,XE        ,ZE        ,XT        , ARCDAT
     *DREF        ,MCHD      ,RHOB       ,QMULT     ,REMAX                  ARCDAT
     * ,FRATE  ,  ARCD(9)                                                   RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      ENTRY EL2010                                                      EL2
      ASSIGN 20 TO IGO                                                  EL2
      GO TO 4                                                           EL2
      ENTRY EL2001                                                      EL2
      ASSIGN 40 TO IGO                                                  EL2
      GO TO 5                                                           EL2
      ENTRY EL2000                                                      EL2
      ASSIGN 50 TO IGO                                                  EL2
```

20 OCT 72 G.01-46

```
      GO TO 5                                        EL2
C                                                    EL2
C                                                    EL2
    4 RAC = ZCGM*COD - XCGM*SID                      EL2
      HAC = -ZCGM*SID - XCGM*COD                     EL2
    5 XEMXCG = XE - XCG                              EL2
      ZEMZCG = ZE - ZCG                              EL2
      FAC = XEMXCG*SID - ZEMZCG*COD                  EL2
      SAC = XEMXCG*COD + ZEMZCG*SID                  EL2
      XJ1 = 1.- XJ                                   EL2
      GO TO IGO                                      EL2
   20 XK2V = XJV*XMCG - XJ1*XMCGV                    EL2
      XK2R = XJR*XMCG - XJ1*XMCGR                    EL2
      XK2M = RAC*T - XJ1*XMCGM                       EL2
   40 XK2T = FAC                                     EL2
      XK2D = T*( XEMXCG*COD + ZEMZCG*SID)            EL2
      XK2A = -XJ1*XMCGA                              EL2
   50 XK2 = FAC*T - XJ1*XMCG                         EL2
C                                                    EL2
      RETURN                                         EL2
      END                                            EL2
```

SUBRØUTINE ENVPRM

Subroutine ENVPRM

Entry Points. ENVPRI

Purpose

ENVPRM computes steepest-descent-solution trajectory environmental parameters that affect vehicle weight. It is used with SSSP sizing synthesis problems only.

Description

Subroutine ENVPRM is called by MODELA at every compute interval of the steepest descent solution trajectory on SSSP type sizing problems.

On branched trajectories having an atmospheric entry portion, it computes maximum dynamic pressure times angle of attack product ($q\alpha$), threshold heat load, and maximum total load factor.

During boost, it computes maximum dynamic pressure and also saves the velocity, altitude flight path angle, time, and Mach number at the point where the maximum dynamic pressure occurs.

Entry point ENVPRI initializes the threshold heat load to zero at the beginning of the trajectory.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DRAG	D	I	Aerodynamic drag (LBS)	/GENF /(	497)	ACCEL	I	DRAG
						BL5	I	DRAG
						BL7	I	DRAG
						BL8	I	DRAG
						ENVPRM	I	DRAG
						FH3	I	DRAG
						OUT	I	DRAG
						PROPB	O	DRAG
						PROPIN	O	DRAG
						SDER3	I	DRAG
						VT	M	DRAG
DVAR	$\dot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
HT1		W	Current heat load above threshold	/ENVPRM/(*	)	ENVPRM	W	HT1
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
INTB		I	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
						BNTG	I	INTB
						ENVPRM	I	INTB
						FNTG	I	INTB
						SDINP	M	INTB
						TEST	I	INTB
						TRAN3	I	INTB
						TRTOSZ	I	INTB
LIFT	L	I	Aerodynamic lift (LBS)	/GENF /(	496)	ACCEL	I	LIFT
						BL4	I	LIFT
						BL5	I	LIFT
						BL6	I	LIFT
						ENVPRM	I	LIFT
						FH3	I	LIFT
						OUT	I	LIFT
						PROPB	O	LIFT
						PROPIN	O	LIFT
						VT	O	LIFT
MACH	M	I	Mach number	/GENF /(*	307)	BERDCO	I	MACH
						ENVPRM	I	MACH
						EQUA3	M	MACH
						OUT	I	MACH

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NSAB		I	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB
Q	q	I	Dynamic pressure (PSF)	/GENF /(	303)	ENVPRM	I	Q
						EQUA3	M	Q
						OUT	I	Q
						PDBC	I	Q
						VT	I	Q
QALF		W	Current maximum dynamic pressure - angle-of-attack product	/ENVPRM/(*	)	ENVPRM	W	QALF
QLIM		M	Saved value of maximum dynamic pressure.	/SIZING/(	264)	ENVPRM	M	QLIM
RAD		I	Radian to angle conversion, 57 29577951	/DATA /(	2)	BEROCD	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADS1	O	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD
SQ		M	A synthesis data array (37,5) that contains the flyback data and some injection quantities	/SIZING/(	74)	ENVPRM	M	SQ
						FLYBKP	M	SQ
						ISPRAT	I	SQ
						PDBC	I	SQ
						PRITVA	I	SQ
						RANGE	M	SQ
						REU3	O	SQ
						SIZE	O	SQ
						SIZEMR	M	SQ
						SIZIN	M	SQ
						STAU	I	SQ
						SUMOUT	M	SQ
						TAMPAR	O	SQ
						TAMPER	M	SQ
						THRUST	M	SQ
						TRTOSZ	M	SQ
						VEHDF	M	SQ
						WTVOL	M	SQ

8 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SQRT		F	Square root function	/SQRT /($	)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTOE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAY02	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
SV		M	A synthesis array (28) containing staging parameters and misc flags	/SIZING/(	46)	ENVPRM	M	SV
						FLYBKP	I	SV
						ITER8	I	SV
						RANGE	I	SV
						SIZEMR	M	SV
						SIZIN	I	SV
						SSSP	I	SV
						SUMOUT	I	SV
						TAMPAR	O	SV
						TAMPER	M	SV
						TRTOSZ	M	SV
						VEHDF	M	SV
						WTVOL	I	SV
THRESH		W	Threshold heating rate	/ENVPRM/(*	)	ENVPRM	W	THRESH
TIME	t	I	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
TLOFAC		W	Current maximum total load factor	/ENVPRM/(*	)	ENVPRM	W	TLOFAC

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
VAR	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	V
						STP3	I	VAR
						TOPM	D	KWOW
						YREF3	M	V
W	W	I	Weight (LBS)	/GENF /(	412)	BL5	I	W
						ENVPRM	I	W
						EQUA3	M	W
						FH3	I	W
						OUT	I	W
						PDBC	I	W
						REU3	I	W
						TRTOSZ	I	W

```
      SUBROUTINE ENVPRM                                                 ENVPRM
C          COMPUTE   ENVIRONMENTAL  PARAMETERS THAT AFFECT  WEIGHT      ENVPRM
      REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                           SIZING
      COMMON /SIZING/                                                   SIZING
C          PHASE II SIZING PARAMERERS                                   SIZING
     *TZ,      VV(3),     QP(14),  EROR,    PZ(5),     VQ,      SW(20), SIZING
     *SV(28),  SQ(37,5),  SE(11),  TLAT,    TLNG,                       SIZING
C         PHASE I SIZING PARAMERERS                                     SIZING
     *WBO,     WLOO,      DWEB,    DWEO,    TOLWT,     WPB,     TWRAT2, SIZING
     *BK1,     BK2,       BK3,     BK4,     ISIZE,     TRAFLG,  TWRATO, SIZING
     *OK1,     OK2,       OK3,     OK4,     PRFLG,     IPASS,   IPSMAX, SIZING
     *AEXIT,   TVACO,     NO,      WFO,     IDVEL,     ISPO,    ISPB,   SIZING
     *XPL,     TVACB,   NNB,       WEO,     WEB,       WO,      WLO,    SIZING
     *DVO,      DVB,       MUB,     MUO,      VSTG,     WPO             SIZING
     *,JTYP,    BECO ,    BSTG ,   ORBI,    ITNBW , ITNOW  ,            SIZING
     * SVDPSQ  , SVDCON  ,IHUNT   ,IOPSTG  ,ISZD(19)                    UH
      COMMON/GENF/                                                      GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED    ,                                            GENF
     *P1        ,P2        ,P3        ,XK1       ,XK2       ,XK3      , GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV        ,PG        ,PP        ,PR        ,PD        ,DPOY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP       ,KPST      ,X         ,KST       ,NAD       ,NCASE    , XCODES
     *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN     ,NEOF ,ILAB(8),JPRP,JGIT,MTT,NPIN(20),JP1,JP2,JP3         XCODES
      COMMON/DATA/                                                      DATA
     *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF     , DATA
     *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4       DATA
      EQUIVALENCE (SE(6),QLIM)                                          ENVPRM
      COMMON/STATE3/                                                    STATE30
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE30
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE30
     *SINPSI    ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE30
     *SYBV   (9),OMEGA     ,OMEGA2    ,                                 STATE30
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE30
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE30
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE30
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE30
     *GDP       ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE30
     *PDO       ,UDO       ,HTDV      ,HTDR                             STATE30
```

```
          REAL MDM , MDV, MDR                                         STATE30
          COMMON/STATE3/                                              STATE30
         *SIN2RO    ,COS2RO     ,COS2GM                               STATE30
          IF(INTB EQ 2)  GO TO  100                                   ENVPRM
C          I        MAXIMUM  DYNAMIC PRESSURE                         ENVPRM
       10 IF( IARC.GE IFIX(SQ(1,3))) RETURN                           ENVPRM
          IF( Q.LE  QLIM    )  RETURN                                 ENVPRM
          QLIM = Q                                                    ENVPRM
          SQ(5,1) = TIME                                              ENVPRM
          SQ(6,1) = VAR(3)                                            ENVPRM
          SQ(7,1) = VAR(1)                                            ENVPRM
          SQ(8,1) = VAR(2)*RAD                                        ENVPRM
          SQ(9,1) = MACH                                              ENVPRM
          RETURN                                                      ENVPRM
C         II  BRANCH  TRAJECTORY TEST                                 ENVPRM
      100 IF(IARC LE NSB) GO TO 10                                    ENVPRM
          IF(IFIX(SQ(1,3)) LE.NSB+NSAB)  GO TO 120                    ENVPRM
          IF(IARC GT.NSB AND IARC.LE.NSB+NSAB) GO TO 200              ENVPRM
          RETURN                                                      ENVPRM
      120 IF(IARC GT NSB+NSAB) GO TO 200                              ENVPRM
          RETURN                                                      ENVPRM
C          III      ENTRY  TRAJECTORY  PARAMETERS  Q*SLPHA AND HEAT RATE ENVPRM
      200 CONTINUE                                                    ENVPRM
          QALF    =Q*APHR                                             ENVPRM
          IF(QALF GT.SV(1)   )    SV(1)  = QALF                       ENVPRM
          TLOFAC =  SQRT(LIFT*LIFT +DRAG*DRAG) / W                    ENVPRM
          IF(TLOFAC GT SV(27)) SV(27)= TLOFAC                         ENVPRM
C     IIIA THRESHOLD   HEAT  LOAD   AND  TIME                         ENVPRM
          IF(THRESH GT 0) GO TO  220                                  ENVPRM
          IF(DVAR(8).LT SQ(30,2))  RETURN                             ENVPRM
          THRESH = TIME                                               ENVPRM
          HT1 =  VAR(8)                                               ENVPRM
          RETURN                                                      ENVPRM
      220 IF(DVAR(8) LT SQ(30,2)) RETURN                              ENVPRM
          SQ(30,3)= TIME -THRESH                                      ENVPRM
          SQ(30,4) =VAR(8) - HT1                                      ENVPRM
          RETURN                                                      ENVPRM
          ENTRY ENVPRI                                                ENVPRM
          THRESH =0.                                                  ENVPRM
          RETURN                                                      ENVPRM
          END                                                         ENVPRM
```

20 OCT 72 G 01-46

SUBRØUTINE
EQUA3

Subroutine EQUA3

Entry Point. EQUA

Purpose

Subroutine EQUA3 computes functions of state only during the integration of the forward steepest descent trajectory and also during the adjoint integration.

Description

The basic logic of this subroutine depends first of all on whether a vacuum simulation, IATM=3, is desired and second whether a forward or adjoint integration is in progress. Secondary logic accounts for the types of table look-ups that are required according to simulation options.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALT	h	I	Altitude	/STATE3/(	3)	EQUA3	I	ALT
						OUT	I	ALT
						PDBC	I	ALT
CD0	C_{D_0}	I	Drag coefficient at $\alpha = 0$	/AEC03 /(	34)	BEROCO	I	CD0
						EQUA3	I	CD0
CD0M	$\partial C_{D_0}/\partial M$	I	See symbol	/AEC03 /(	35)	BEROCO	I	CD0M
						EQUA3	I	CD0M
CLA	C_{L_α}	M	Lift coefficient slope	/AEC03 /(	50)	BEROCO	M	CLA
						EQUA3	M	CLA
						VT	I	CLA
CLAM	$\partial C_{L_\alpha}/\partial M$	M	See symbol	/AEC03 /(	48)	BEROCO	I	CLAM
						EQUA3	M	CLAM
CL0	C_{L_0}	I	Lift coefficient at $\alpha = 0$	/AEC03 /(	36)	BEROCO	I	CL0
						EQUA3	I	CL0
CL0M	$\partial C_{L_0}/\partial M$	I	See symbol	/AEC03 /(	40)	BEROCO	I	CL0M
						EQUA3	I	CL0M
CMA	C_{M_α}	M	Moment coefficient slope	/AEC03 /(	42)	EQUA3	M	CMA
						MAMECO	I	CMA
						VT	I	CMA
CMAM	$\partial C_{M_\alpha}/\partial M$	M	See symbol	/AEC03 /(	43)	EQUA3	M	CMAM
						MAMECO	I	CMAM
CM0	C_{M_0}	I	Moment coefficient at $\alpha = 0$	/AEC03 /(	45)	EQUA3	I	CM0
						MAMECO	I	CM0
CM0M	$\partial C_{M_0}/\partial M$	I	See symbol	/AEC03 /(	46)	EQUA3	I	CM0M
						MAMECO	I	CM0M
COSGAM	$\cos(\gamma)$	O	See symbol	/STATE3/(	687)	ACCEL	I	COSGAM
						BL4	I	COSGAM
						BL8	I	COSGAM
						DER3A	I	COSGAM
						EQUA3	O	COSGAM
						MODELA	I	COSGAM
						MODELB	I	COSGAM
						OUT	I	COSGAM
						PDBC	I	COSGAM
						PDY3A	I	COSGAM
COSPSI	$\cos(\psi)$	O	See symbol	/STATE3/(	705)	BL4	I	COSPSI
						BL7	I	COSPSI
						BL8	I	COSPSI
						DER3A	I	COSPSI
						EQUA3	O	COSPSI
						MODELA	I	COSPSI
						MODELB	I	COSPSI
						PDBC	I	COSPSI
						PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	M	See symbol	/STATE3/(	707)	BL4	I	COSRHO
						BL7	I	COSRHO
						BL8	I	COSRHO
						DER3A	I	COSRHO
						EQUA3	M	COSRHO
						MODELA	I	COSRHO
						MODELB	I	COSRHO
						OUT	I	COSRHO
						PDBC	I	COSRHO
						PDY3A	I	COSRHO
CS	a	M	Speed of sound (FT/SEC)	/GENF /(	310)	EQUA3	M	CS
						OUT	I	CS
CSR		I	Deriv Of speed of sound wrt alt.	/GENF /(	314)	EQUA3	I	CSR

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DB	D_b	I	Base drag (LBS)	/GENF /(	537)	ACCEL	I	DB
						BL4	I	DB
						BL6	I	DB
						BL7	I	DB
						BL8	I	DB
						EQUA3	I	DB
						FH3	I	DB
						OUT	I	DB
						SDER3	I	DB
						VT	I	DB
DBR		I	Partial of base drag wrt altitude	/GENF /(	536)	ACCEL	I	DBR
						BL4	I	DBR
						BL6	I	DBR
						BL7	I	DBR
						BL8	I	DBR
						EQUA3	I	DBR
						FH3	I	DBR
						VT	I	DBR
DZM	P_a	M	Atmospheric pressure (PSF)	/GENF /(	308)	EQUA3	M	DZM
						FH2	I	PA
						IMPUL	I	PA
						OUT	I	PA
						PDBC	I	PA
						SDER3	I	PA
ER	E_R	I	Earth radius. (FT)	/GLOBAL/(	2)	COORDS	I	ER
						CRASH	I	REM
						EQUA3	I	ER
						GEINP	I	ER
						PADS1	I	ER
						PDBC	I	ER
						SOMG	I	ER
						TRTOSZ	I	ER
FK	k	I	Induced drag coefficient	/AECO3 /(	37)	BEROCD	I	FK
						EQUA3	I	FK
FKM	$\partial k/\partial M$	I	See symbol	/AECO3 /(	47)	BEROCD	I	FKM
						EQUA3	I	FKM
FRATE		I	Input rated vacuum thrust per engine (LBS)	/ARCDAT/(	42)	EQUA3	I	FRATE
						FXDAT	I	FRATE
						PROPB	I	FRATE
						PROPIN	I	FRATE
						SIZIN	O	FRATE
FVAC		M	Total vacuum thrust [rocket] (LBS)	/GENF /(	528)	ACCEL	I	FVAC
						EQUA3	M	FVAC
						FH2	I	FVAC
						IMPUL	M	FVAC
						PROPB	M	FVAC
						PROPIN	M	FVAC
						SDER3	I	FVAC
G	g	M	Gravitational attraction (FT/SEC**2)	/GENF /(	301)	BL4	I	G
						BL7	I	G
						BL8	I	G
						DER3A	I	G
						EQUA3	M	G
						MODELA	I	G
						MODELB	I	G
						PDY3A	I	G
						SDER3	I	G
						SDINP	M	G
GAM	γ	I	Relative flight path angle (RAD)	/STATE3/(	2)	EQUA3	I	GAM
						GUI3A	I	GAM
						OUT	I	GAM
GH		O	Partial of gravity wrt altitude	/GENF /(	563)	BL7	I	GH
						BL8	I	GH
						EQUA3	O	GH
						PDY3A	I	DGDH

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
GR	g_r	I	Gravitational acceleration at surface of the earth (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
IATM		I	Atmosphere option flag	/ARCDAT/(	7)	EQUA3	I	IATM
						FXDAT	I	IATM
						OUT	I	IATM
						PDBC	I	IATM
						VT	I	IATM
IFOB		I	Forward or adjoint integration flag = 1 means forward = 2 means adjoint	/XCODES/(	178)	ACCEL	I	IFOB
						BEROCO	I	IFOB
						BLYNE	I	IFOB
						EQUA3	I	IFOB
						IMPUL	I	IFOB
						SPLYNE	I	IFOB
						TOPM	O	IFOB
JAER		I	Aerodynamic model option flag	/ARCDAT/(	9)	BEROCO	I	JAER
						EQUA3	I	JAER
						GEINP	I	JAER
						OUT	I	JAER
						PROPB	I	JAER
						PROPIN	I	JAER
						VT	I	JAER
JPRO		I	Propulsion model option flag	/ARCDAT/(	10)	EQUA3	I	JPRO
						GEINP	I	JPRO
						IMPUL	I	JPRO
						MODELA	I	JPRO
						PROPB	I	JPRO
						PROPIN	I	JPRO
JPRP		I	Propulsion flag for different rocket options	/XCODES/(	194)	ACCEL	I	JPRP
						DER3A	I	JPRP
						EQUA3	I	JPRP
						MODELA	I	JPRP
						PDY3A	I	JPRP
						PROPB	O	JPRP
						PROPIN	O	JPRP
M	m	I	Mass	/STATE3/(	4)	ACCEL	I	M
						BL4	I	M
						BL8	I	M
						EQUA3	I	M
						OUT	I	M
						SDER3	I	M
MACH	M	M	Mach number	/GENF /(	307)	BEROCO	I	MACH
						ENVPRM	I	MACH
						EQUA3	M	MACH
						OUT	I	MACH
MACHR		O	Partial of mach number wrt altitude	/GENF /(	524)	EQUA3	O	MACHR
						PROPB	O	ZERO
						PROPIN	O	ZERO
						VT	I	MACHR
MACHV		O	Partial of mach number wrt velocity	/GENF /(	525)	EQUA3	O	MACHV
						VT	I	MACHV

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
MAEA		I	Curve number		/ARCDAT/(	18)	EQUA3 GEINP PROPB PROPIN	I I I I	MAEA MAEA MAEA MAEA
MAEB		I	Curve number		/ARCDAT/(	19)	EQUA3	I	MAEB
MAEC		I	Curve number		/ARCDAT/(	20)	EQUA3	I	MAEC
MAED		I	Curve number		/ARCDAT/(	21)	EQUA3	I	MAED
MAEE		I	Curve number		/ARCDAT/(	22)	EQUA3	I	MAEE
MAEF		I	Curve number		/ARCDAT/(	23)	EQUA3	I	MAEF
MAEG		I	Curve number		/ARCDAT/(	24)	EQUA3	I	MAEG
MDB		I	Curve number - base drag table		/ARCDAT/(	31)	EQUA3	I	MDB
MTT		I	Thrust curve number		/XCODES/(	196)	EQUA3 PROPB PROPIN	I M M	MTT MTT MTT
MXCG		I	Curve number -xcg table		/ARCDAT/(	27)	EQUA3	I	MXCG
MZCG		I	Curve number- zcg table		/ARCDAT/(	28)	EQUA3	I	MZCG
OCORHO	$\omega\times$COSRHO	M	See symbol		/STATE3/(	708)	DER3A EQUA3 MODELA MODELB PDBC PDY3A	I M I I I I	OCORHO OCORHO OCORHO OCORHO OCORHO OCORHO
OCORO2	$\omega\times$DCORHO	O	See symbol		/STATE3/(	709)	DER3A EQUA3 MODELA MODELB PDY3A	I O I I I	OCORO2 OCORO2 OCORO2 OCORO2 OCORO2
OMGZ	ω	I	Earth rotation rate	(RAD/SEC)	/GLOBAL/(	3)	ADID3A CRASH DER3A EQUA3 GEINP MODELA MODELB PDBC PDY3A SDINP TOPM	I I I I I I I I I I I	OMGZ OMEGA OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ
PSI	ψ	I	Azimuth		/STATE3/(	5)	EQUA3 GUI3A OUT	I I I	PSI PSI PSI
Q	q	M	Dynamic pressure	(PSF)	/GENF /(	303)	ENVPRM EQUA3 OUT PDBC VT	I M I I I	Q Q Q Q Q
QR		M	Partial of dynamic pressure wrt altitude		/GENF /(	526)	EQUA3 VT	M I	QR QR
QS		O	Product of dynamic pressure and aero Ref. Area	(LBS)	/GENF /(	304)	EQUA3 VT	O I	QS QS
QV		M	Partial of dynamic pressure wrt velocity		/GENF /(	527)	EQUA3 TEST VT	M M I	QV QV QV
R	R	M	Radial distance from earth center to vehicle	(FT)	/GENF /(	305)	BL4 BLT BL8 DER3A EQUA3 MODELA MODELB PDBC PDY3A TRTOSZ	I I I I M I I I I I	R R R R R R R R R R

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
RAD		I	Radian to angle conversion, 57.29577951	/DATA /(	2)	BEROCO	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADSI	O	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD
RHO	ρ	I	Latitude	/STATE3/(	6)	EQUA3	I	RHO
						OUT	I	RHO
RO	ρ_a	I	Atmospheric density (SLUGS/FT**3)	/GENF /(	309)	BL7	I	RO
						BL8	I	RO
						DER3A	I	RO
						EQUA3	I	RO
						OUT	I	RO
						PDBC	I	RO
						PDY3A	I	RO
ROR		I	Deriv. Of density wrt alt.	/GENF /(	313)	BL7	I	ROR
						BL8	I	ROR
						EQUA3	I	ROR
						PDBC	I	ROR
						PDY3A	I	ROR
SINGAM	$\sin(\gamma)$	O	See symbol	/STATE3/(	688)	BL4	I	SINGAM
						BL7	I	SINGAM
						BL8	I	SINGAM
						DER3A	I	SINGAM
						EQUA3	O	SINGAM
						MODELA	I	SINGAM
						MODELB	I	SINGAM
						PDBC	I	SINGAM
						PDY3A	I	SINGAM
						SDER3	I	SINGAM
SINPSI	$\sin(\psi)$	O	See symbol	/STATE3/(	704)	BL4	I	SINPSI
						BL7	I	SINPSI
						BL8	I	SINPSI
						DER3A	I	SINPSI
						EQUA3	O	SINPSI
						MODELA	I	SINPSI
						MODELB	I	SINPSI
						PDBC	I	SINPSI
						PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	O	See symbol	/STATE3/(	706)	BL4	I	SINRHO
						BL7	I	SINRHO
						BL8	I	SINRHO
						DER3A	I	SINRHO
						EQUA3	O	SINRHO
						MODELA	I	SINRHO
						MODELB	I	SINRHO
						OUT	I	SINRHO
						PDBC	I	SINRHO
						PDY3A	I	SINRHO
SREF	S_{ref}	I	Aerodynamic reference area (FT^2)	/ARCDAT/(	1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SDINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
T	T	O	Thrust	(LBS)	/GENF /(	411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T
TAIRB		I	Airbreather thrust.	(LBS)	/AIRBRE/(	1)	EQUA3	I	TAIRB
							FH4	I	TAIRB
TBURN	t_b	I	Rocket burn initiation time on forward trajectory[sd]		/GENF /(	499)	EQUA3	I	TBURN
							MODELA	I	TBURN
							PROPB	O	TBURN
							PROPIN	M	TBURN
TIME	t	I	Time (elapsed)		/GENF /(	493)	ADICB3	O	TIME
							AST3	I	TIME
							BNTG	M	TIME
							CON3	I	TIME
							DTF3	I	TIME
							ENVPRM	I	TIME
							EQUA3	I	TIME
							FNTG	M	TIME
							MODELA	I	TIME
							OUT	I	TIME
							PDBC	I	TIME
							PROPIN	I	TIME
							REU3	M	TIME
							RKTA3A	M	TT
							RKTB3A	M	TT
							YREF3	M	TIME
TIMEPH	τ_p	O	Phase time	(SEC)	/GENF /(	318)	EQUA3	O	TIMEPH
							FNTG	M	TIMEPH
							GETIT	I	TIMEPH
							GUI3A	I	TIMEPH
							OUT	I	TIMEPH
TIMES	τ	O	Arc time	(SEC)	/GENF /(	319)	AST3	I	TIMES
							EQUA3	O	TIMES
							FNTG	M	TIMES
							GETIT	I	TIMES
							OUT	I	TIMES
TMULT	T_{mult}	I	Thrust multiplier or number of engines		/ARCDAT/(	4)	EQUA3	I	TMULT
							FXDAT	M	TMULT
							PROPB	I	TMULT
							PROPIN	I	TMULT
							SIZIN	O	TMULT
TOP		I	Elasped time at phase initiation		/GENF /(	320)	BNTG	M	TOP
							EQUA3	I	TOP
							FNTG	M	TOP
TOS		I	Elasped time at arc initiation		/GENF /(	321)	BNTG	M	TOS
							EQUA3	I	TOS
							FNTG	M	TOS

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBR	CODE	VAR
V	v	I	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	V
							STP3	I	VAR
							TOPM	D	KWOW
							YREF3	M	V
W	W	M	Weight	(LBS)	/GENF /(	412)	BL5	I	W
							ENVPRM	I	W
							EQUA3	M	W
							FH3	I	W
							OUT	I	W
							PDBC	I	W
							REU3	I	W
							TRTOSZ	I	W
XCG	X_{CG}	I	Center of gravity body x station	(FT)	/GENF /(	558)	EL2	I	XCG
							EQUA3	I	XCG
							VT	I	XCG
XCGM	$\partial X_{CG}/\partial m$	O	See symbol		/AECO3 /(	38)	EL2	I	XCGM
							EQUA3	O	XCGM
							VT	I	XCGM
XJ	J	I	Control blend factor		/GENF /(	560)	EL2	I	XJ
							EQUA3	I	XJ
							OUT	I	XJ
							VT	I	XJ
XJR		O	Partial of blend factor wrt altitude		/GENF /(	562)	EL2	I	XJR
							EQUA3	O	XJR
							VT	I	XJR
XJV		O	Partial of blend factor wrt velocity		/GENF /(	561)	EL2	I	XJV
							EQUA3	O	XJV
							VT	I	XJV
ZCG	Z_{CG}	I	Center of gravity body z station	(FT)	/GENF /(	559)	EL2	I	ZCG
							EQUA3	I	ZCG
							VT	I	ZCG
ZCGM	$\partial Z_{CG}/\partial m$	O	See symbol		/AECO3 /(	39)	EL2	I	ZCGM
							EQUA3	O	ZCGM
							VT	I	ZCGM

```
      SUBROUTINE EQUA3                                                  EQUA3
C                                                                       COMN
C       COMPUTES FUNCTIONS  OF  STATE ONLY                              COMN
C       SKIPS  STATE  PARTIALS  ON  FORWARD TRAJECTORY (IFOB=1)         COMN
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI    ,COSPSI    ,SINRHO    ,COSRHO    ,DCORHO    ,DCORO2    , STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                 STATE3D
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP       ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO       ,UDO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                      STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,  EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2), EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD), EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8), HTD),(DVAR(9),SQ2D)           EQUV3
      REAL M,MU,MD                                                      EQUV3
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      , ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      , ARCDAT
     *XLMAX     ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      , ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      , ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      , ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        , ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                 ARCDAT
     * ,FRATE   , ARCD(9)                                               RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      COMMON/GLOBAL/                                                    GLOBAL
     *GR        ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM        GLOBAL
     *,JJOP(10) ,IFATAL    ,NARC      ,NBRAN     ,NFARC     ,ID(4)      GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                               GLOBAL
     *,GM       ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)      GLOBAL
     *,ITPSO ,KSOL     ,KGLOBL(8)                                       RETAP
      DIMENSION DZM(8)                                                  EQUA3
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
     *I2OP      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       , XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
     *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
     *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         , XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN     ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/DATA/                                                      DATA
     *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      , DATA
     *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4        DATA
      COMMON/GENF/                                                      GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *OTS       ,DT        ,G         ,OPSQ      ,Q         ,QS        , GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                  GENF
```

```
      *XJV        ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP        , GENF
      *FRATED     ,IRATED     ,                                                 GENF
      *P1         ,P2         ,P3         ,XK1        ,XK2        ,XK3        , GENF
      *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D       , GENF
      *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V       , GENF
      *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P       , GENF
      *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O       , GENF
      *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M       , GENF
      *PV         ,PG         ,PP         ,PR         ,PO         ,OPDY(3,8)  , GENF
       REAL  LIFTR , LIFT , LIFTA,LIFTM ,                  MACH, MACHR,         GENF
      * ISP, ISPF, MACHV,LIFTV , IRATED                                         FRAT
       DIMENSION TPH1(10),TST1(10)                                              GENF
       EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                       GENF
       COMMON/AECO3/                                                            AECO3
      *APHO       ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA        , AECO3
      *SINA       ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI     , AECO3
      *COSPHI     ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDO        , AECO3
      *CDOM       ,CLO        ,FK         ,XCGM       ,ZCGM       ,CLOM       , AECO3
      *CM         ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM       ,FKM        , AECO3
      *CLAM       ,CL         ,CLA        ,CLM        ,                         AECO3
      *CD         ,CDA        ,CDM                                              AECO3
       EQUIVALENCE (DZM,PA)                                                     EQUA3
       COMMON/AIRBRE/ TAIRB,TAIRBV,TAIRBH,SFC,SFCV,SFCH                         FIXED
       EQUIVALENCE(JIN,ILAB(1))                                                 PO14
       ENTRY EQUA                                                               EQUA3
C      THIS  PROGRAM  CALC. STATE  FUNCTIONS  ONLY                              EQUA3
       R= ER+ ALT                                                               EQUA3
       SINGAM= SIN(GAM)                                                         EQUA3
       COSGAM= COS(GAM)                                                         EQUA3
       W=M*GR                                                                   EQUA3
       TIMEPH = TIME -TOP                                                       EQUA3
       TIMES= TIME -TOS                                                         EQUA3
       G=  GR*(ER/ R)**2                                                        EQUA3
       SINPSI= SIN(PSI)                                                         EQUA3
       COSPSI= COS(PSI)                                                         EQUA3
       SINRHO= SIN(RHO)                                                         EQUA3
       COSRHO= COS(RHO)                                                         EQUA3
       OCORHO= OMGZ*COSRHO                                                      EQUA3
       OCOR02= OCORHO*OMGZ                                                      EQUA3
C  I    IF ZERO AERO  OR  VACUUM  SKIP  ATMOSPHERE  CALCULATION TO VI           COMN
C                                                                               COMN
       IF(SREF EQ.0) GO TO 120                                                  PO14
       IF(IATM-1) 10,20,120                                                     PO14
    10 CALL  ANLATM (ALT,DZM,IFOB-1)                                            EQUA3
       GO TO 30                                                                 EQUA3
    20 CALL PAT63(ALT,DZM,IFOB)                                                 EQUA3
    30 Q =    5* RO*V*V                                                         EQUA3
       QS=  Q*SREF                                                              EQUA3
       MACH= V/CS                                                               EQUA3
       IF(IFOB EQ.1) GO TO 40                                                   EQUA3
       VSQ = V* V                                                               EQUA3
       MACHV= 1./CS                                                             EQUA3
       MACHR= -V/CS/CS*CSR                                                      EQUA3
       GH = -2.*G/R                                                             EQUA3
       QR=VSQ*ROR/2.                                                            EQUA3
       QV=  RO*V                                                                EQUA3
C                                                                               COMN
C       II    DO  APPROPRIATE  TABLE  LOOKUPS FOR AERODYNAMIC MODELS            COMN
    40 GO TO (70,80,50) , JAER                                                  EQUA3
    50 CONTINUE                                                                 EQUA3
       CALL SPLYNE(MAEE,MACH,CMA,CMAM)                                          EQUA3
       CALL SPLYNE(MAEF,MACH,CMO,CMOM)                                          EQUA3
       CALL SPLYNE(MAEG,Q ,XJ , XJQ)                                            EQUA3
    60 CALL SPLYNE(MXCG,W ,XCG, XCGW)                                           EQUA3
       CALL SPLYNE(MZCG,W ,ZCG, ZCGW)                                           EQUA3
       CMA= CMA *RAD                                                            EQUA3
       IF(IFOB.EQ 1) GO TO 70                                                   EQUA3
       CMAM=CMAM*RAD                                                            EQUA3
       XJV=XJQ*QV                                                               APR
       XJR=XJQ*QR                                                               APR
       XCGM   = XCGW*GR                                                         EQUA3
       ZCGM   = ZCGW*GR                                                         EQUA3
```

```
   70 CONTINUE                                                        EQUA3
      CALL SPLYNE(MAEA,MACH,CLA,CLAM)                                 EQUA3
      CALL SPLYNE(MAEB,MACH,CLO,CLOM)                                 EQUA3
      CALL SPLYNE(MAEC,MACH,CDO,CDOM)                                 EQUA3
      CALL SPLYNE(MAED,MACH,FK,FKM)                                   EQUA3
      CLA = CLA*RAD                                                   EQUA3
      IF(IFOB.EQ.1) GO TO  80                                         EQUA3
      CLAM= CLAM*RAD                                                  EQUA3
   80 CONTINUE                                                        FIXED
C                                                                     COMN
C       III       PROPULSION  CALCULATION                             COMN
C                                                                     COMN
C       III-A    AIR BREATHER  THRUST  AND  SFC.                      COMN
      IF(JPRO.NE.2) GO  TO  85                                        FIXED
      CALL  BLYNE(V,ALT, TAIRB)                                       FIXED
      T= TAIRB                                                        FIXED
      GO TO 110                                                       FIXED
   85 CONTINUE                                                        FIXED
C                                                                     COMN
C       III-B     ROCKET                                              COMN
C                                                                     COMN
C             TEST  FOR  ROCKET MODE  (THROTTLED OR UNTHROTTLED)      COMN
C         JPRP =1 CONST, 2 UNTHROTTLED , 3 THROTTLED                  COMN
      IF(JPRP-2) 110,100,90                                           POW
   90 IF(JIN.GT.0) GO TO  110                                         EQUA3
C                                                                     COMN
C        III-C   IF NO THRUST TABLE  GO TO IV, ELSE INTERPOLATE VAC THRST COMN
C                                                                     COMN
  100 IF(MTT.EQ.0) GO TO 109                                          PHISZ
      TPRP = TIME - TBURN                                             FRAT
      CALL  SPLYNE (MTT, TPRP,FVAC,DUM)                               EQUA3
      FVAC = FVAC* TMULT                                              EQUA3
      GO TO 110                                                       PHISZ
C                                                                     COMN
C       IV    IF  RATED THRUST ZERO GO TO V ,ELSE  COMPUTE VAC  THRUST COMN
  109 IF(FRATE.LE.0.) GO TO 110                                       PHISZ
      FVAC=FRATE*TMULT                                                PHISZ
C                                                                     COMN
C       V      INTERPOLATE  BASE  DRAG                                COMN
  110 CALL SPLYNE( MDB,ALT,DB,DBR)                                    EQUA3
      RETURN                                                          EQUA3
C                                                                     COMN
C       VI     VACUUM  CALCULATIONS   THEN GO TO IIIB                 COMN
  120 DO 130 I=1,8                                                    EQUA3
  130 DZM(I)=0.                                                       EQUA3
      CS=1000.                                                        EQUA3
      Q=0.                                                            EQUA3
      GH=-2.*G/R                                                      PHISZ
      QS=0.                                                           EQUA3
      MACH=20.                                                        EQUA3
      IF(JAER.NE.3) GO TO 80                                          EQUA3
  140 CALL SPLYNE(MXCG,W ,XCG, XCGW)                                  EQUA3
      CALL SPLYNE(MZCG,W ,ZCG, ZCGW)                                  EQUA3
      IF(IFOB.EQ.1)GO TO 80                                           EQUA3
      XCGM = XCGW*GR                                                  EQUA3
      ZCGM = ZCGW*GR                                                  EQUA3
      GH= -2.*G/R                                                     PO14
      GO TO 80                                                        EQUA3
      END                                                             EQUA3
```

20 OCT 72 6.01-46

SUBRØUTINE
FH1

Subroutine FM1

<u>Entry Points</u>. FH1001, FH1000

Purpose

Subroutine FH1 is the governing equation (XK1) associated with the thrust element in the in-plane control vector when the thrust is a constant.

Description

The solution of the in-plane control using the governing equations is described in Section 9 of Volume I. This subroutine is called from BLGCØN.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CT		I	Constant value of vacuum thrust	(LBS)	/GENF	/(551)	FH1	I	CT
T	T	I	Thrust	(LBS)	/GENF	/(411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T
XK1		O	First control vector governing equation value corresponds to error in thrust		/GENF	/(572)	BLGCON	I	XK1
							FH1	O	XK1
							FH2	O	XK1
							FH3	O	XK1
							FH4	O	XK1
							MODELA	I	XK1
XK1T		O	Partial of governing equation wrt state or control vector component		/GENF	/(575)	BLGCON	I	XK1T
							FH1	O	XK1T
							FH2	O	XK1T
							FH3	M	XK1T
							FH4	O	XK1T
							MODELA	I	XK1T

```
      SUBROUTINE FH1                                                    FH1
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TQL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPO       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV         XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED     ,IRATED    ,                                           GENF
     *P1         ,P2        ,P3        ,XK1       ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                   MACH, MACHR, GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,VD(9)     ,SVV(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15), STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE3D
     *SVBV    (9),OMEGA     ,OMEGA2    ,                                STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                            STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO      COS2RO    ,COS2GM                                     STATE3D
      ENTRY FH1001                                                      FH1
   40 XK1T = 1.                                                         FH1
      ENTRY FH1000                                                      FH1
   50 XK1 = T - CT                                                      FH1
C                                                                       FH1
      RETURN                                                            FH1
      END                                                               FH1
```

SUBRØUTINE
FH2

Subroutine FH2

Entry Points. FH2010, FH2001, FM2000

Purpose

Subroutine FH2 is the governing equation (XK1) associated with the thrust element in the in-plane control vector when the thrust is modeled as a vacuum thrust modified by a nozzle back-pressure correction.

Description

The solution of the in-plane control using the governing equations is described in Section 9 of Volume I. This subroutine is called from BLGCØN.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AE	A_{exit}	I	Total nozzle exit area		/GENF	/(520)	ACCEL	I	AE
							FH2	I	AE
							IMPUL	I	AE
							PROPB	O	AE
							PROPIN	O	AE
							SDER3	I	AE
FVAC		I	Total vacuum thrust [rocket]	(LBS)	/GENF	/(528)	ACCEL	I	FVAC
							EQUA3	M	FVAC
							FH2	I	FVAC
							IMPUL	M	FVAC
							PROPB	M	FVAC
							PROPIN	M	FVAC
							SDER3	I	FVAC
PA	P_a	I	Atmospheric pressure	(PSF)	/GENF	/(308)	EQUA3	M	DZM
							FH2	I	PA
							IMPUL	I	PA
							OUT	I	PA
							PDBC	I	PA
							SDER3	I	PA
PAR		I	Deriv. Of press Wrt alt.		/GENF	/(312)	ACCEL	I	PAR
							FH2	I	PAR
T	T	I	Thrust	(LBS)	/GENF	/(411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T
XK1		O	First control vector governing equation value corresponds to error in thrust		/GENF	/(572)	BLGCON	I	XK1
							FH1	O	XK1
							FH2	O	XK1
							FH3	O	XK1
							FH4	O	XK1
							MODELA	I	XK1
XK1R		O	Partial of governing equation wrt state or control vector component		/GENF	/(593)	FH2	O	XK1R
							FH3	M	XK1R
							FH4	O	XK1R
XK1T		O	Partial of governing equation wrt state or control vector component		/GENF	/(575)	BLGCON	I	XK1T
							FH1	O	XK1T
							FH2	O	XK1T
							FH3	M	XK1T
							FH4	O	XK1T
							MODELA	I	XK1T

```
      SUBROUTINE FH2                                                    FH2
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARO(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COT1(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,OPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED     ,                                            GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                MACH, MACHR,   GENF
     * ISP, ISPF, MACHV,LIFTV  , IRATED                                 FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15), STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE3D
     *SVBV    (9),OMEGA     ,OMEGA2    ,                                 STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM  , MDV, MDR                                              STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RD     ,COS2RD    ,COS2GM                                     STATE3D
      ENTRY FH2010                                                      FH2
      XK1R = AE*PAR                                                     FH2
      ENTRY FH2001                                                      FH2
   40 XK1T = 1.                                                         FH2
      ENTRY FH 2000                                                     FH2
   50 XK1 = T - FVAC + AE*PA                                            FH2
C                                                                       FH2
      RETURN                                                            FH2
      END                                                               FH2
```

20 OCT 72 G 01-46

SUBRØUTINE FH3

Subroutine FH3

Entry Point. FH3010, FH3001, FH3000

Purpose

Subroutine FH3 is the governing equation (XK1) associated with the thrust element in the in-plane control vector when the thrust is being throttled to maintain a limit total acceleration.

Description

The solution of the in-plane control using the governing equations is described in Section 9 of Volume I. This subroutine is called from BLGCØN.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
CODAE	$\cos(\alpha-\delta_E)$	I	See symbol		/GENF	/(549)	ACCEL	I	CODAE
							BL4	I	CODAE
							BL6	I	CODAE
							BL7	I	CODAE
							BL8	I	CODAE
							FH3	I	CODAE
							SDER3	I	CODAE
							VT	O	CODAE
COSA	$\cos\alpha$	I	See symbol		/AECO3	/(8)	ACCEL	I	COSA
							BL4	I	COSA
							BL6	I	COSA
							BL7	I	COSA
							BL8	I	COSA
							FH3	I	COSA
							OUT	I	COSA
							VT	M	COSA
DB	D_b	I	Base drag	(LBS)	/GENF	/(537)	ACCEL	I	DB
							BL4	I	DB
							BL6	I	DB
							BL7	I	DB
							BL8	I	DB
							EQUA3	I	DB
							FH3	I	DB
							OUT	I	DB
							SDER3	I	DB
							VT	I	DB
DBR		I	Partial of base drag wrt altitude		/GENF	/(536)	ACCEL	I	DBR
							BL4	I	DBR
							BL6	I	DBR
							BL7	I	DBR
							BL8	I	DBR
							EQUA3	I	DBR
							FH3	I	DBR
							VT	I	DBR
DRAG	D	I	Aerodynamic drag	(LBS)	/GENF	/(497)	ACCEL	I	DRAG
							BL5	I	DRAG
							BL7	I	DRAG
							BL8	I	DRAG
							ENVPRM	I	DRAG
							FH3	I	DRAG
							OUT	I	DRAG
							PROPB	O	DRAG
							PROPIN	O	DRAG
							SDER3	I	DRAG
							VT	M	DRAG
DRAGA		I	Partial of drag wrt angle of attack		/GENF	/(534)	ACCEL	I	DRAGA
							BL5	I	DRAGA
							BL7	I	DRAGA
							BL8	I	DRAGA
							FH3	I	DRAGA
							VT	M	DRAGA
DRAGR		I	Partial drag wrt altitude		/GENF	/(533)	ACCEL	I	DRAGR
							BL5	I	DRAGR
							BL7	I	DRAGR
							BL8	I	DRAGR
							FH3	I	DRAGR
							VT	M	DRAGR
DRAGV		I	Partial of drag wrt velocity		/GENF	/(532)	ACCEL	I	DRAGV
							BL5	I	DRAGV
							BL7	I	DRAGV
							BL8	I	DRAGV
							FH3	I	DRAGV
							VT	M	DRAGV
GMAX	G_{MAX}	I	Maximum total acceleration g load		/ARCDAT	/(12)	BL5	I	GMAX
							FH3	I	GMAX
							MODELA	I	GMAX
							PROPB	I	GMAX
							PROPIN	I	GMAX

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
GR	g_r	I	Gravitational acceleration at surface of the earth (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
LIFT	L	I	Aerodynamic lift (LBS)	/GENF /(	496)	ACCEL	I	LIFT
						BL4	I	LIFT
						BL5	I	LIFT
						BL6	I	LIFT
						ENVPRM	I	LIFT
						FH3	I	LIFT
						OUT	I	LIFT
						PROPB	O	LIFT
						PROPIN	O	LIFT
						VT	O	LIFT
LIFTA		I	Partial of lift wrt angle-of-attack	/GENF /(	531)	ACCEL	I	LIFTA
						BL4	I	LIFTA
						BL5	I	LIFTA
						BL6	I	LIFTA
						FH3	I	LIFTA
						VT	O	LIFTA
LIFTM		I	Partial of LIFT wrt mass	/GENF /(	535)	ACCEL	I	LIFTM
						BL4	I	LIFTM
						BL5	I	LIFTM
						BL6	I	LIFTM
						FH3	I	LIFTM
						VT	O	LIFTM
LIFTR		I	Partial of lift wrt altitude	/GENF /(	530)	ACCEL	I	LIFTR
						BL4	I	LIFTR
						BL5	I	LIFTR
						BL6	I	LIFTR
						FH3	I	LIFTR
						VT	O	LIFTR
LIFTV		I	Partial of lift wrt velocity	/GENF /(	529)	ACCEL	I	LIFTV
						BL4	I	LIFTV
						BL5	I	LIFTV
						BL6	I	LIFTV
						FH3	I	LIFTV
						VT	O	LIFTV
SIDAE	$\sin(\alpha-\delta_E)$	I	See symbol	/GENF /(	557)	ACCEL	I	SIDAE
						BL4	I	SIDAE
						BL6	I	SIDAE
						BL7	I	SIDAE
						BL8	I	SIDAE
						FH3	I	SIDAE
						VT	O	SIDAE
SINA	$\sin\alpha$	I	See symbol	/AEC03 /(	7)	ACCEL	I	SINA
						BL4	I	SINA
						BL6	I	SINA
						BL7	I	SINA
						BL8	I	SINA
						FH3	I	SINA
						GUI3A	M	SINA
						OUT	I	SINA
						VT	M	SINA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
T	T	I	Thrust (LBS)	/GENF	/(411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	0	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	0	T
						PROPIN	0	T
						REU3	0	T
						SDER3	I	T
W	W	I	Weight (LBS)	/GENF	/(412)	BL5	I	W
						ENVPRM	I	W
						EQUA3	M	W
						FH3	I	W
						OUT	I	W
						PDBC	I	W
						REU3	I	W
						TRTOSZ	I	W
XK1		0	First control vector governing equation value corresponds to error in thrust	/GENF	/(572)	BLGCON	I	XK1
						FH1	0	XK1
						FH2	0	XK1
						FH3	0	XK1
						FH4	0	XK1
						MODELA	I	XK1
XK1A		M	Partial of governing equation wrt state or control vector component	/GENF	/(581)	BLGCON	I	XK1A
						FH3	M	XK1A
XK1D		M	Partial of governing equation wrt state or control vector component	/GENF	/(578)	BLGCON	I	XK1D
						FH3	M	XK1D
XK1M		M	Partial of governing equation wrt state or control vector component	/GENF	/(602)	FH3	M	XK1M
XK1R		M	Partial of governing equation wrt state or control vector component	/GENF	/(593)	FH2	0	XK1R
						FH3	M	XK1R
						FH4	0	XK1R
XK1T		M	Partial of governing equation wrt state or control vector component	/GENF	/(575)	BLGCON	I	XK1T
						FH1	0	XK1T
						FH2	0	XK1T
						FH3	M	XK1T
						FH4	0	XK1T
						MODELA	I	XK1T
XK1V		M	Partial of governing equation wrt state or control vector component	/GENF	/(584)	BLGCON	I	XK1V
						FH3	M	XK1V
						FH4	0	XK1V

```
      SUBROUTINE FH3                                                    FH3
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF       ,EJ         ,XISP       ,TMULT      ,DTNC       ,DTPI       , ARCDAT
     *IATM       ,IMODE      ,JAER       ,JPRO       ,QMAX       ,GMAX       , ARCDAT
     *XLMAX      ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX      ,MAEA       , ARCDAT
     *MAEB       ,MAEC       ,MAED       ,MAEE       ,MAEF       ,MAEG       , ARCDAT
     *MT         ,MISP       ,MXCG       ,MZCG       ,MWDA       ,MWDB,      , ARCDAT
     *MDB        ,XCGR       ,ZCGR       ,XE         ,ZE         ,XT         , ARCDAT
     *DREF       ,MCND       ,RHOB       ,QMULT      ,REMAX                    ARCDAT
     * ,FRATE   , ARCO(9)                                               RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)    ,TOL(9)     ,SVAR(10)   ,WDC(20)    , GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)    ,COTI(9,9)  ,DCON(9)    ,DTP        , GENF
     *DTS        ,DT         ,G          ,DPSQ       ,Q          ,QS         , GENF
     *R          ,RE         ,MACH       ,PA         ,RO         ,CS         , GENF
     *VNU        ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ      , GENF
     *SVSQ       ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)      , GENF
     *TST(20)    ,TPH   (20),DIS(20)    ,DIP(20)    ,T          ,W          , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)   ,DIS1(20)   ,TIME       ,OMP        , GENF
     *TIMPR      ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)    , GENF
     *AE         ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV      , GENF
     *QR         ,QV         ,FVAC       ,LIFTV      ,                         GENF
     *LIFTR      ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA      , GENF
     *            LIFTM      ,DBR        ,DB         ,ISP        ,ISPF       , GENF
     *                        ULFT       ,ULFTV      ,ULFTR      ,ULFTA      , GENF
     *XMCG       ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE      , GENF
     *CULFT      ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SIO        , GENF
     *COD        ,SIDAE      ,XCG        ,ZCG        ,XJ                       GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP        , GENF
     *FRATED    ,IRATED     ,                                           GENF
     *P1         ,P2         ,P3        ,XK1        ,XK2        ,XK3        , GENF
     *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D       , GENF
     *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V       , GENF
     *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P       , GENF
     *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O       , GENF
     *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M       , GENF
     *PV         ,PG         ,PP         ,PR         ,PO         ,DPDY(3,8)  , GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)      ,SVY(10)    , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15)  , STATE3D
     *SINPSI     ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2     , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2     ,                                   STATE3D
     *VDV        ,GDV        ,RDV        ,MDV        ,PDV        ,ODV        , STATE3D
     *UDV        ,VDG        ,GDG        ,RDG        ,PDG        ,ODG        , STATE3D
     *UDG        ,VDR        ,GDR        ,MDR        ,PDR        ,ODR        , STATE3D
     *UDR        ,VDM        ,GDM        ,MDM        ,PDM        ,VDP        , STATE3D
     *GDP        ,PDP        ,ODP        ,UDP        ,VDO        ,GDO        , STATE3D
     *PDO        ,UDO        ,HTDV       ,HTDR                                 STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                    STATE3D
      COMMON/GLOBAL/                                                    GLOBAL
     *GR         ,ER         ,OMGZ      ,XLAMRF     ,VMURF      ,LUM        GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC        ,NBRAN     ,NFARC      ,ID(4)      GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                                 GLOBAL
     *,GM         PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)        GLOBAL
     *,ITPSO ,KSOL      ,KGLOBL(8)                                         RETAP
      COMMON/AECO3/                                                     AECO3
     *APHO       ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA        , AECO3
     *SINA       ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI     , AECO3
     *COSPHI     ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDG        , AECO3
     *CDOM       ,CLO        ,FK         ,XCGM       ,ZCGM       ,CLOM       , AECO3
     *CM         ,CMA        ,CMAR ,CMM  ,CMO        ,CMDM       ,FKM        , AECO3
     *CLAM       ,CL         ,CLA        ,CLM        ,                         AECO3
     *CD         ,CDA        ,CDM                                              AECO3
      ASF(X, Y) = T1*X + T2*Y                                           FH3
C                                                                       FH3
      ENTRY FH3010                                                      FH3
```

```
      ASSIGN 20 TO IGO                                                  FH3
      GO TO 4                                                           FH3
      ENTRY FH3001                                                      FH3
      ASSIGN 40 TO IGO                                                  FH3
      GO TO 5                                                           FH3
      ENTRY FH3000                                                      FH3
      ASSIGN 50 TO IGO                                                  FH3
      GO TO 5                                                           FH3
C                                                                       FH3
    4 T5 = -DRAGR - DBR*COSA                                            FH3
      T6 = LIFTR - DBR*SINA                                             FH3
    5 TCDAE = T*CODAE                                                   FH3
      TSDAE = T*SIDAE                                                   FH3
      DBCA = DB*COSA                                                    FH3
      DBSA = DB*SINA                                                    FH3
      ZZ = -TSDAE + DBSA                                                FH3
      YY = -TCDAE + DBCA                                                FH3
      T1 = -YY - DRAG                                                   FH3
      T2 = LIFT - ZZ                                                    FH3
      T3 = ZZ - DRAGA                                                   FH3
      T4 = LIFTA - YY                                                   FH3
      GO TO IGO                                                         FH3
C                                                                       FH3
   20 XK1V = ASF(-DRAGV, LIFTV)                                         FH3
      XK1V = XK1V + XK1V                                                FH3
      XK1R = ASF(T5, T6)                                                FH3
      XK1R = XK1R + XK1R                                                FH3
      XK1M = T2*LIFTM - GR*GMAX**2*W                                    FH3
      XK1M = XK1M + XK1M                                                FH3
   40 XK1T = ASF(CODAE, SIDAE)                                          FH3
      XK1T = XK1T + XK1T                                                FH3
      XK1D = ASF(TSDAE, -TCDAE)                                         FH3
      XK1D = XK1D + XK1D                                                FH3
      XK1A = ASF(T3, T4)                                                FH3
      XK1A = XK1A + XK1A                                                FH3
   50 XK1 = ASF(T1, T2) - (GMAX*W)**2                                   FH3
      RETURN                                                            FH3
      END                                                               FH3
```

SUBRØUTINE FH4

Subroutine FH4

Entry Points. FH4010, FH4001, FH4000

Purpose

Subroutine FH4 is the governing equation (XK1) associated with the thrust element in the in-plane control vector when the airbreathing propulsion option is used.

Description

The solution of the in-plane control is described in Section 9 of Volume I. This subroutine is called from BLGCØN.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
T	T	I	Thrust	(LBS)	/GENF /(	411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T
TAIRB		I	Airbreather thrust.	(LBS)	/AIRBRE/(	1)	EQUA3	I	TAIRB
							FH4	I	TAIRB
TAIRBH		I	Partial of airbreather thrust WRT altitude		/AIRBRE/(	3)	ACCEL	I	TAIRBH
							FH4	I	TAIRBH
TAIRBV		I	Partial of airbreather thrust WRT velocity		/AIRBRE/(	2)	ACCEL	I	TAIRBV
							FH4	I	TAIRBV
XK1		O	First control vector governing equation value corresponds to error in thrust		/GENF /(	572)	BLGCON	I	XK1
							FH1	O	XK1
							FH2	O	XK1
							FH3	O	XK1
							FH4	O	XK1
							MODELA	I	XK1
XK1R		O	Partial of governing equation wrt state or control vector component		/GENF /(	593)	FH2	O	XK1R
							FH3	M	XK1R
							FH4	O	XK1R
XK1T		O	Partial of governing equation wrt state or control vector component		/GENF /(	575)	BLGCON	I	XK1T
							FH1	O	XK1T
							FH2	O	XK1T
							FH3	M	XK1T
							FH4	O	XK1T
							MODELA	I	XK1T
XK1V		O	Partial of governing equation wrt state or control vector component		/GENF /(	584)	BLGCON	I	XK1V
							FH3	M	XK1V
							FH4	O	XK1V

```
      SUBROUTINE  FH4                                                    FH4
      COMMON/GLOBAL/                                                     GLOBAL
     *GR          ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM       GLOBAL
     *,JJOP(10)   ,IFATAL    ,NARC      ,NBRAN     ,NFARC     ,ID(4)     GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                                GLOBAL
     *,GM         ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)     GLOBAL
     *,ITPSO  ,KSOL     ,KGLOBL(8)                                       RETAP
      COMMON/GENF/                                                       GENF
     *OMG(20)     ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WOC(20)  , GENF
     *A(9,9)      ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS         ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R           ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU         ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ        ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)     ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)    ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR       ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE          ,FP        ,FPOLO     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR          ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR       ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *             LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                        ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG        ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT       ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD         ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                   GENF
     *XJV         ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED      ,                                           GENF
     *P1          ,P2        ,P3        ,XK1       ,XK2       ,XK3      , GENF
     *XK1T        ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A        ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G        ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R        ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U        ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV          ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,               MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                        GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                 GENF
      COMMON/AIRBRE/ TAIRB,TAIRBV,TAIRBH,SFC,SFCV,SFCH                   FH4
      ENTRY FH4010                                                       FH4
      XK1R =-TAIRBH                                                      FH4
      XK1V =-TAIRBV                                                      FH4
C                                                                        FH4
      ENTRY FH4001                                                       FH4
      XK1T = 1.                                                          FH4
C                                                                        FH4
      ENTRY FH4000                                                       FH4
      XK1 = T  - TAIRB                                                   FH4
      RETURN                                                             FH4
      END                                                                FH4
```

10 OCT 72 6 01-46

435

SUBRØUTINE FNTG

FNTG

Purpose

FNTG controls integration of forward trajectory and all subsidiary functions.

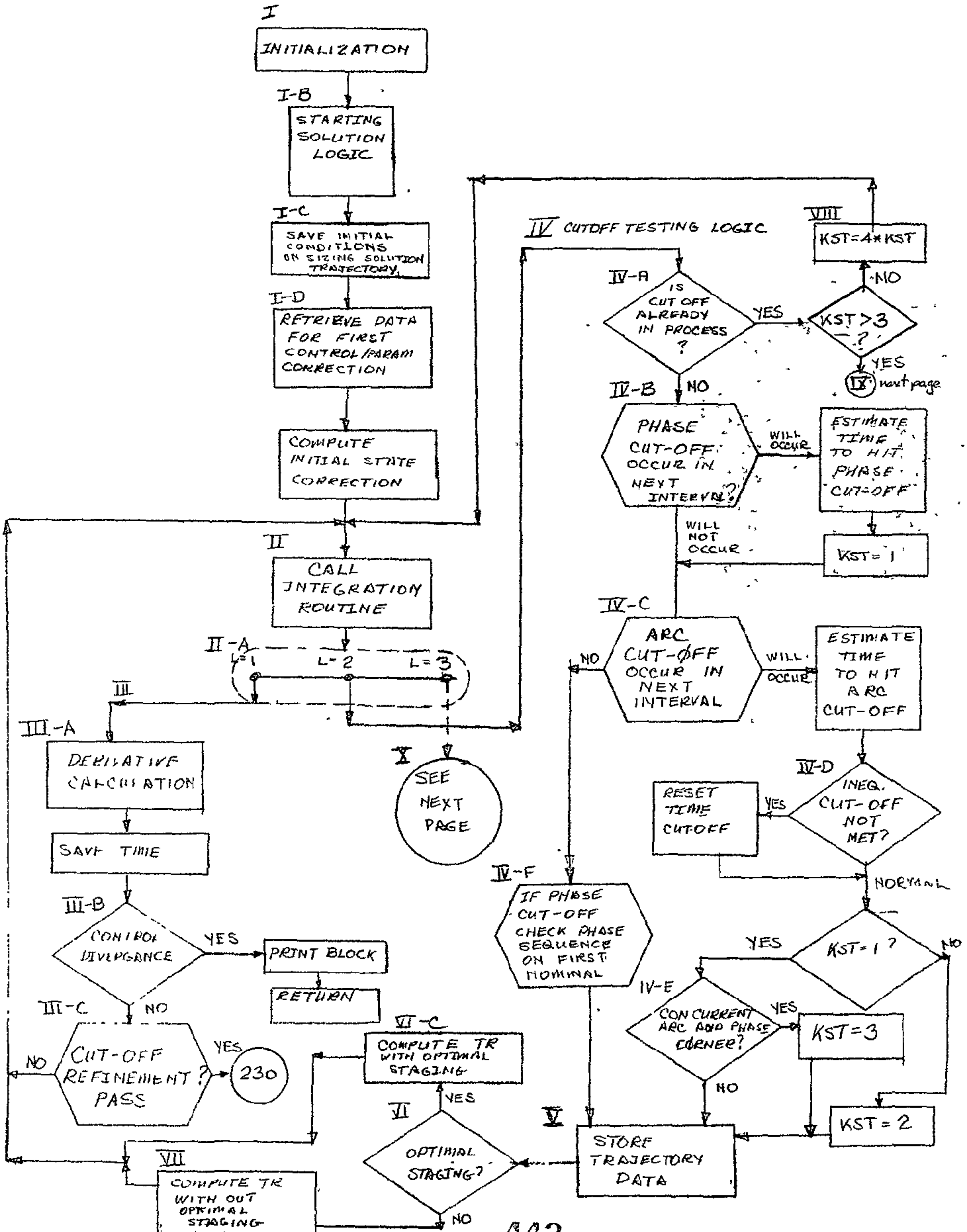
Description

FNTG is a subroutine in the UNIVAC version of the program, however, in the CDC version of PADS it is a "Program" which heads up the largest overlay in the steepest descent module.

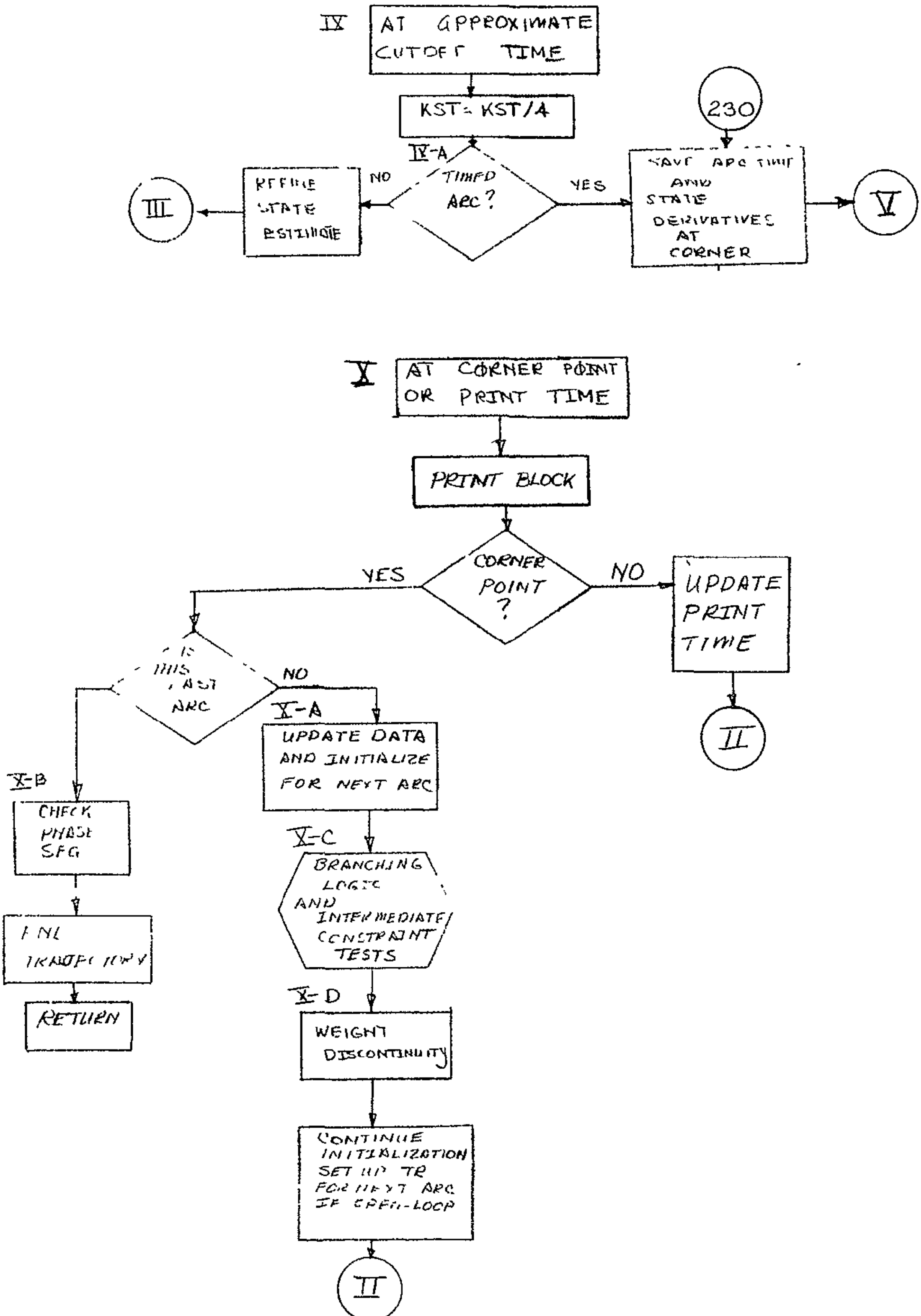
The logic in this program is geared mainly to the Runge-Kutta numerical integration subroutine, RKUTTA. Through the use of the traffic flag, L. The meanings of L are listed below.

L = 1,	Evaluate derivatives
L = 2,	Check cut-off and store data
L = 3,	At corner point or print time
L 4,	Restart integration.

FNTG


I
INITIALIZATION
I-B
STARTING SOLUTION LOGIC
I-C
SAVE INITIAL CONDITIONS ON SIZING SOLUTION TRAJECTORY
I-D
RETRIEVE DATA FOR FIRST CONTROL/PARAM CORRECTION
COMPUTE INITIAL STATE CORRECTION
II
CALL INTEGRATION ROUTINE
II-A
L=1
L=2
L=3
III
III-A
DERIVATIVE CALCULATION
SAVE TIME
III-B
CONTROL DIVERGANCE
YES
PRINT BLOCK
RETURN
NO
III-C
CUT-OFF REFINEMENT PASS
YES
230
NO
X
SEE NEXT PAGE
IV CUTOFF TESTING LOGIC
IV-A
IS CUT OFF ALREADY IN PROCESS ?
YES
KST>3 ?
NO
VIII
KST=4*KST
YES
IX next page
IV-B
NO
PHASE CUT-OFF OCCUR IN NEXT INTERVAL?
WILL OCCUR
ESTIMATE TIME TO HIT PHASE CUT-OFF
KST=1
WILL NOT OCCUR
IV-C
ARC CUT-OFF OCCUR IN NEXT INTERVAL
NO
WILL OCCUR
ESTIMATE TIME TO HIT ARC CUT-OFF
IV-D
INEQ. CUT-OFF NOT MET?
YES
RESET TIME CUTOFF
NORMAL
KST=1 ?
YES
NO
IV-F
IF PHASE CUT-OFF CHECK PHASE SEQUENCE ON FIRST NOMINAL
IV-E
CONCURRENT ARC AND PHASE CORNER?
YES
KST=3
NO
KST=2
V
STORE TRAJECTORY DATA
VI
OPTIMAL STAGING?
YES
VI-C
COMPUTE TR WITH OPTIMAL STAGING
NO
VII
COMPUTE TR WITH OUT OPTIMAL STAGING

FNTG (CONTINUED)



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ALPHA	α	O	Angle of attack (RAD)	/AEC03 /(	3)	BEROCO	I	ALPHA
						BLGCON	M	ALPHA
						BL2	I	ALPHA
						FNTG	O	ALPHA
						MAMECO	I	ALPHA
						MODELA	M	ALPHA
						MODELB	O	ALPHA
						REU3	O	ALPHA
						VT	I	ALPHA
APHO	α_{old}	I	Angle of attack from last nominal trajectory (DEG)	/AEC03 /(	1)	AST3	M	APHO
						FNTG	I	APHO
						MTX3A	I	APHO
						OUT	I	APHO
						PROPB	O	AEZRO
						PROPIN	O	AEZRO
ARCDA	S_{ref}	I	Aerodynamic reference area (FT^2)	/ARCDAT/(	1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SDINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF
DIP		O	Array of phase end integration intervals for trial trajectory	/GENF /(	391)	BNTG	I	DIP
						FNTG	O	DIP
DIS		O	Array of arc end integration intervals for trial trajectory	/GENF /(	371)	BNTG	I	DIS
						FNTG	O	DIS
DT		M	Integration interval (SEC)	/GENF /(	300)	BNTG	M	DT
						FNTG	M	DT
						REU3	I	DT
						RKTA3A	I	P
						RKTB3A	I	P
						STP3	I	DT
						YREF3	O	DT
DTNC	$\Delta\tau$	I	Integration interval (SEC)	/ARCDAT/(	5)	BNTG	I	DTNC
						FNTG	I	DTNC
						GEINP	M	DTNC
						PROPIN	I	DTNC
DTP		M	Altered integration interval reqd to hit phase cut off [sd] (SEC)	/GENF /(	298)	FNTG	M	DTP
DTPI		I	Print frequency for trajectory	/ARCDAT/(	6)	FNTG	I	DTPI
						GEINP	M	DTPI
DTS		M	Altered integration interval reqd to hit arc cut-off [sd] (SEC)	/GENF /(	299)	FNTG	M	DTS
IARC		M	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
ICOR		I	Phase sequence array	/XCODES/(	10)	ADJUST FNTG PRMSET SDINP	I I I M	ICOR ICOR ICOR ICOR
IND		M	Flag indicates whether on first nominal trajectory (IND=1)	/XCODES/(	141)	AST3 BGET3 FNTG GUI3A MTX3A PROPIN	I I M I I I	IND IND IND IND IND IND
INTB		I	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A BNTG ENVPRM FNTG SDINP TEST TRAN3 TRTOSZ	I I I I M I I I	INTB INTB INTB INTB INTB INTB INTB INTB
IOPEN		M	Closed to open-loop control switch when equal to 2	/XCODES/(	142)	FNTG PROPIN	M O	IOPEN IOPEN
IPFLG1		I	IPFLG1≠0 supresses print-out of velocity losses and inertial Euler angles.	/GLOBAL/(	69)	FNTG OUT PDBC PRINT TRTOSZ	I I I I O	IPFLG1 IPFLG1 IPFLG1 IPFLG1 IPFLG1
IPH		M	Phase number	/XCODES/(	143)	ADID3A ADJUST AST3 BNTG FNTG GETIT GUI3A SDINP	I I I M M I I M	IPH IPH IPH IPH IPH IPH IPH IPH
IPST		M	Phase counter for first nominal trajectory	/XCODES/(	167)	AST3 FNTG GUI3A REU3	I M I I	IPST IPST IPST IPST
ISPH		O	Sign of phase cut-off	/XCODES/(	144)	FNTG STP3	O I	ISPH ISPH
ISST		O	Sign of arc cut-off	/XCODES/(	145)	FNTG STP3	O I	ISST ISST
ISTART		I	Initialization and divergance flag	/XCODES/(	147)	AST3 BLGCON BLYNE FNTG MODELA PROPIN REU3 TEST TOPM	O O O I O O I M M	ISTART ISTART ISTART ISTART ISTART ISTART ISTART ISTART ISTART
ISTOP		M	Arc cut-off flag	/XCODES/(	175)	FNTG STP3	M I	ISTOP ISTOP
ISTPP		M	Phase cut-off flag	/XCODES/(	176)	FNTG STP3	M I	ISTPP ISTPP
ITER		I	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3 FNTG GETIT MODELA OUT PAYO2 PROPIN TEST TOPM	I I I I I M I M M	ITER ITER ITER ITER ITER ITER ITER ITER ITER
ITI		I	Optimized arc time flag	/XCODES/(	30)	ADJUST FNTG SDINP	M I O	ITI ITI ITI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ITT		W	Arc number when derivatives were last evaluated	/FNTG /(*	)	FNTG	W	ITT
IVAR		M	Cut-off variable option indicator	/XCODES/(	150)	FNTG	M	IVAR
						STP3	O	IVAR
						TOL3	I	IVAR
JGID		I	Control option flag array	/XCODES/(	32)	BNTG	I	JGID
						FNTG	I	JGID
						SDINP	M	JGID
JGII		M	Control option	/XCODES/(	195)	ACCEL	I	JGII
						BNTG	O	JGII
						DER3A	I	JGII
						FNTG	M	JGII
						GUI3A	I	JGII
						MODELA	I	JGII
						MODELB	I	JGII
						MTX3A	I	JGII
						PDY3A	I	JGII
JPH		M	Phase cut-off option flag	/XCODES/(	72)	BNTG	I	JPH
						FNTG	M	JPH
						SDINP	M	JPH
JPS		M	Absolute value of phase cut-off option code	/XCODES/(	152)	ADID3A	I	JPS
						BNTG	M	JPS
						FNTG	M	JPS
						STP3	I	JPS
						TOL3	I	JPS
JS		M	Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
						ADIC3A	I	JS
						ADID3A	I	JS
						BNTG	M	JS
						FNTG	M	JS
						PROPB	I	JS
						PROPIN	I	JS
						STP3	I	JS
						TOL3	I	JS
JST		I	Arc cut-off option flag	/XCODES/(	112)	ADICB3	I	JST
						BNTG	I	JST
						FNTG	I	JST
						SDINP	M	JST
K		O	Storage retrieval flag indicates end of arc, phase, or data.	/XCODES/(	156)	AST3	O	K
						FNTG	O	K
						GETIT	M	K
						MODELA	I	K
						SDINP	M	K
KPST		M	Controls logic for compute interval during adjoint integration	/XCODES/(	155)	BNTG	M	KPST
						FNTG	M	KPST
KSOL		I	An internal flag that has the same significance as ITPSO.	/GLOBAL/(	94)	FNTG	I	KSOL
						PADS1	O	KSOL
						SDINP	M	KSOL
KST		M	Arc or phase cut-off flag	/XCODES/(	157)	ADJUST	I	KST
						FNTG	M	KST
L		M	Integration traffic control flag L = 1 means evaluate derivatives = 2 check cut-off = 3 print or cut-off detected	/XCODES/(	177)	BNTG	M	L
						FNTG	M	L
						OUT	I	L
						RKTA3A	M	L
						RKTB3A	M	L
						SDINP	M	L
LUM		I	Program control flag. LUM = 0. Steepest descent only; LUM = 1 Steepest descent and adjoint transformation stored on tape, LUM = 2 Steepest descent and QL, LUM = 3 QL only.	/GLOBAL/(	6)	AST3	I	LUM
						FNTG	I	LUM
						GEINP	I	LUM
						PADS1	M	LUM
						SDINP	I	LUM
						TOPM	M	LUM
NARC	N_3	I	Number of subarcs in the problem.	/GLOBAL/(	18)	FNTG	I	NARC
						GEINP	M	NARC
						PROPIN	I	NARC
						SDINP	I	NARC
						SIZIN	I	NARC

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NPA		I	Running count of number of adjustable parameters to be perturbed on remainder of trajectory	/PARAM /(	14)	ADJUST	M	NPAR
						FNTG	I	NPA
						MTX3A	I	NPA
						TOPM	D	NPA
NPARA		I	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	D	NPARA
NPH		O	Number of phases in trajectory	/XCODES/(	164)	BNTG	I	NPH
						FNTG	O	NPH
						PRMSET	I	NPH
						SDINP	M	NPH
						TEST	I	NPH
						TOPM	I	NPH
NSAB		I	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB
NST		O	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST
OMG	Ω_j	I	Array of arc cut off values [sd]	/GENF /(	1)	ADJUST	M	OMG
						FNTG	I	OMG
						PRMSET	M	OMG
						PROPB	I	OMG
						SDINP	M	OMG
						STP3	I	OMG
						TOPM	D	IOMG
OMGP		M	Array of phase cut off values [sd]	/GENF /(	21)	ADJUST	O	OMGP
						FNTG	M	OMGP
						PRMSET	O	OMGP
						SDINP	M	OMGP
OMP		M	Phase cut-off value	/GENF /(	494)	ADJUST	O	OMP
						FNTG	M	OMP
						STP3	I	OMP

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
RAD		I	Radian to angle conversion, 57.29577951	/DATA	/(2)	BEROCO	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADS1	D	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD
RDI		I	Angle to radian conversion, .01745329252	/DATA	/(3)	BLICO	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	D	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI
STT		W	Last time value when derivatives were evaluated	/FNTG	/(*)	FNTG	W	STT
SVAR	$y\|_{t=0}$	I	Array of state values at initial problem time [sd]	/GENF	/(79)	ADJUST	O	SVAR
						BNTG	I	SVAR
						FNTG	I	SVAR
						PRMSET	M	SVAR
						REU3	I	SVAR
						SDINP	M	SVAR
						TEST	I	SVAR
						TOPM	I	SVAR
						TRTOSZ	I	SVAR
TIME	t	M	Time (elapsed)	/GENF	/(493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
TIMEPH	τ_p	M	Phase time (SEC)	/GENF	/(318)	EQUA3	O	TIMEPH
						FNTG	M	TIMEPH
						GETIT	I	TIMEPH
						GUI3A	I	TIMEPH
						OUT	I	TIMEPH
TIMES	τ	M	Arc time (SEC)	/GENF	/(319)	AST3	I	TIMES
						EQUA3	O	TIMES
						FNTG	M	TIMES
						GETIT	I	TIMES
						OUT	I	TIMES
TIMPR		M	Trajectory print time	/GENF	/(495)	BNTG	O	TIMPR
						FNTG	M	TIMPR
						RKTA3A	I	TP
						RKTB3A	I	TP
TOP		M	Elasped time at phase initiation	/GENF	/(320)	BNTG	M	TOP
						EQUA3	I	TOP
						FNTG	M	TOP

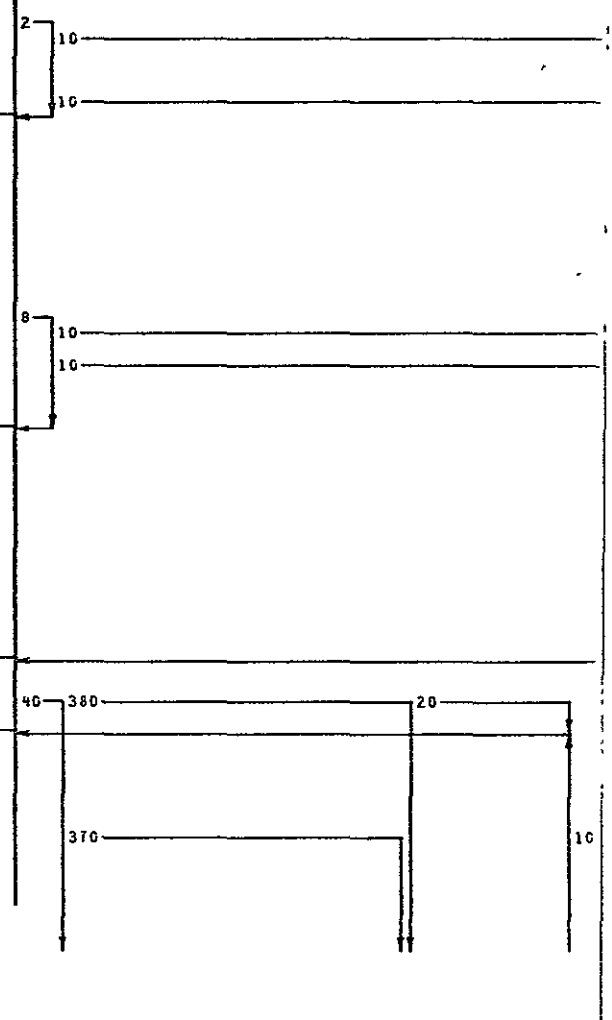
FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
TOS		M	Elasped time at arc initiation	/GENF	/(321)	BNTG	M	TOS
						EQUA3	I	TOS
						FNTG	M	TOS
TPH		O	Array of phase end times on trial trajectory [sd]	/GENF	/(351)	FNTG	O	TPH
						TEST	I	TPH
TST		O	Array of arc end times on trial trajectory [sd]	/GENF	/(331)	ADICB3	I	TST
						BNTG	I	TST
						FNTG	O	TST
						TEST	I	TST
WORK		I	Working array, contains TOPEN1, TOPEN2, and PHIWT	/STS	/(3)	ADEQ3A	I	WORK
						FNTG	I	WORK
						MODELB	O	WORK
						MTX3A	I	WORK
						SDINP	M	WORK
						TEST	M	WORK

```
      PROGRAM FNTG                                                      FNTG
C         CONTROLS  INTEGRATION OF  FORWARD  TRAJECTORY                 COMN
C         AND  ALL  SUBSIDIARY  FUNCTIONS                               COMN
C      USES  TRAFFIC FLAG L  =1 CALC  DERIV.  ,2 TEST FOR CUT-OFF,      COMN
C                             3 PRINT OR  CORNERS,4  RESTART            COMN
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE         ,FP        ,FPOLD     ,FPO       ,MACHR     ,MACHV     , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED     ,IRATED    ,                                           GENF
     *P1         ,P2        ,P3        ,XK1       ,XK2       ,XK3       , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,               MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/AECO3/                                                     AECO3
     *APHO       ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       , AECO3
     *SINA       ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI    , AECO3
     *COSPHI     ,GDPN      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,COO       , AECO3
     *CDOM       ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      , AECO3
     *CM         ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       , AECO3
     *CLAM       ,CL        ,CLA       ,CLM       ,                       AECO3
     *CD         ,CDA       ,CDM                                        AECO3
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)   , STATE3D
     *XL(9,9)    ,YOP(20,9),YOS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    , STATE3D
     *SVBV    (9),OMEGA     ,OMEGA2    ,                                  STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                     STATE3D
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF       ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      , ARCDAT
     *IATM       ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      , ARCDAT
     *XLMAX      ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      , ARCDAT
```

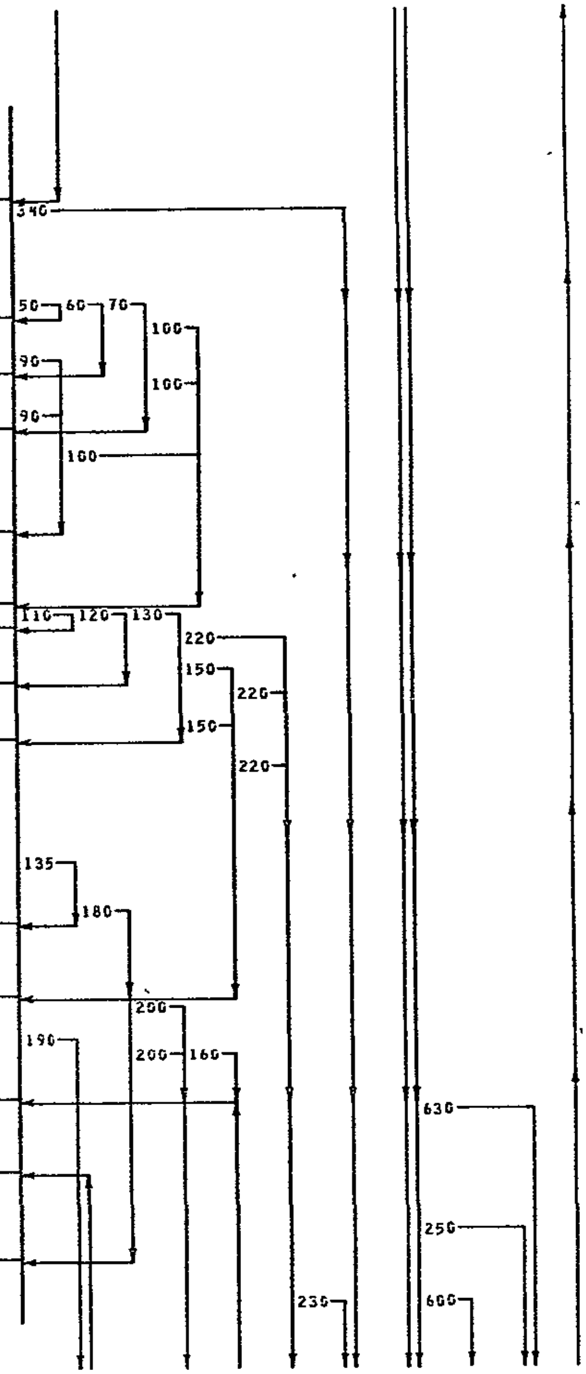
```
C     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      , ARCDAT
      *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      , ARCDAT
      *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        , ARCDAT
      *OREF      ,MCND      ,RHOB      ,OMULT     ,REMAX                  ARCDAT
      * ,FRATE   , ARCD(9)                                                RETAP
       DIMENSION  ARCDA(40)                                               ARCDAT
       EQUIVALENCE(SREF,ARCDA)                                            ARCDAT
       COMMON/GLOBAL/                                                     GLOBAL
      *GR        ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM         GLOBAL
      *,JJOP(10) ,IFATAL    ,NARC        ,NBRAN   ,NFARC     ,ID(4)       GLOBAL
      *,KTAB(20),ITAB(20)  ,SIG   ,MAXTAB                                 GLOBAL
      *,GM       ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)       GLOBAL
      *,ITPSO ,KSOL     ,KGLOBL(8)                                        RETAP
       COMMON/STS/                                                        STS
      *DPAY      ,PMIN      ,WORK (20),NWDS      ,IPC    (7),NITER      , STS
      *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)            STS
       COMMON/DATA/                                                       DATA
      *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      , DATA
      *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4        DATA
       DIMENSION SS(10)                                                   FNTG
       COMMON/PARAM/                                                      PARAM
      *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),            PARAM
      *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                        PARAM
      *,DELP(9)                                                           PARAM
       REAL MUB, MUD, ISPB, ISPO, IDVEL,NNB,ND                            SIZING
       COMMON /SIZING/                                                    SIZING
C           PHASE II SIZING PARAMERERS                                    SIZING
      *TZ,      VV(3),     QP(14),  EROR,    PZ(5),     VQ,      SW(20),  SIZING
      *SV(28),  SQ(37,5),  SE(11),  TLAT,     TLNG,                       SIZING
C          PHASE I SIZING PARAMERERS                                      SIZING
      *WBO,     WLOO,     OWEB,     OWEO,     TOLWT,    WPB,     TWRAT2,  SIZING
      *BK1,     BK2,      BK3,      BK4,      ISIZE,    TRAFLG,  TWRATO,  SIZING
      *OK1,     OK2,      OK3,      OK4,      PRFLG,    IPASS,   IPSMAX,  SIZING
      *AEXIT,   TVACO,    ND,       WFO,      IDVEL,    ISPO,    ISPB,    SIZING
      *XPL,     TVACB,   NNB,       WEO,      WEB,      WO,      WLO,     SIZING
      *DVO,      DVB,      MUB,       MUO,     VSTG,     WPO                SIZING
      *,JTYP,    BECO ,   BSTG ,    ORBI,    ITNBW , ITNOW  ,             SIZING
      * SVDPSQ  , SVDCON  ,IHUNT   ,IOPSTG  ,ISZD(19)                     UH
C                                                                       . COMN
C        I      INITIALIZATION                                            COMN
C               SPECIAL  FLAGS  USED                                      COMN
C               K  STORAGE RETRIEVAL FLAG =1 NORMAL ,2 PHASE END          COMN
C                                           3 ARC END, 4 END OF DATA      COMN
C             IND  =1 ON STARTING NOMINAL  TRAJ.                          COMN
C                  =2 ON TRIAL OR SOLUTION TRAJ.                          COMN
C             KST  =0  DURING NORMAL  INTEGRATION                         COMN
C                  =1  DURING PHASE  CUT-OFF                              COMN
C                  =2  DURING ARC  CUT-OFF                                COMN
C                  =3  DURING CONCURRENT ARC AND PHASE CUT-OFF            COMN
C           ISTART =  1  ON  STARTING  NOM.                               COMN
C                  =  2  ON  TRIAL                                        COMN
C                  =  6  MEANS CONTROL COMPUTATION  DIVERGANCE            COMN
C                        OR EXCESSIVE  CHANGE IN  STAGING TIME            COMN
C                       (CAUSES  HALVING  DOWN )                          COMN
C           IOPEN =1     CLOSED  LOOP  CONTROL                            COMN
C                 =2     OPEN  LOOP  CONTROL                              COMN
       STT=-RAD                                                           FNTG
       K=1                                                                FNTG
C           INIT.  FORWARD  STORAGE                                       COMN
       CALL BEGWR                                                         FNTG
       IND =1                                                             FNTG
       IF(ISTART NE 1) IND=2                                              FNTG
C           CLOSED  LOOP  FLAG AND SWITCHING  TIME  SET                   COMN
       IOPEN =1                                                           FNTG
       TOPEN=WORK(1)                                                      FNTG
       IPH =1                                                             FNTG
       IPST = 1                                                           FNTG
       JPS = IABS(JPH(IPH,IND))                                           FNTG
       ISTPP=JPS-2                                                        FNTG
       ISPH= ISIGN(1,JPH(IPH,IND))                                        FNTG
       OMP = OMGP(IPH,IND)                                                FNTG
       JGII=JGID(IPH,IND)                                                 FNTG
       TOP =  SVAR(1)                                                     FNTG
       KST =0                                                             FNTG
       ICOP=1                                                             FNTG
```

20 OCT 72 G.01-46

```
      TIME =TOS= SVAR(1)                                           FNTG
      CALL  EQIN                                                   FNTG
C           INIT   STATE  AND  AUXILIARY QUANTITIES                COMN
      IARC=1                                                       FNTG
C           GET   DATA  FOR  FIRST ARC                             COMN
      CALL READMS(9,ARCDA,51,IARC)                                 FRAT
C           SET   ARC  CUT-OFF  FLAGS , COMPUTE  INTERVAL  ETC     COMN
      JS= IABS(JST(IARC))                                          FNTG
      ISTOP=JS-2                                                   FNTG
      ISST= ISIGN(1,JST(IARC))                                     FNTG
      DT=DTNC                                                      FNTG
      L=4                                                          FNTG
      TIMPR =TIME                                                  FNTG
C           CALL  ARC  INITIALIZATION  LOGIC                       COMN
      CALL PROPIN                                                  FNTG
      KPST = 1                                                     FNTG
C       I-A    TEST  FOR  STARTING  NOM. TRAJ. AND  IF  CONTROL    COMN
C              IS  ON  FILE 11, IF NOT  STARTING  NOM. GO  TO  I-B COMN
C              IF STARTING NOM. BUT NO  CONTROL ON FILE  GO TO II, COMN
C              ELSE  INIT  STORAGE RETRIEVAL  AND  GO TO II        COMN
      IF(IND.NE.1) GO TO 2                                         RETAP
      IF(KSOL.EQ. 0) GO TO 10                                      RETAP
      CALL  PHIN                                                   RETAP
      CALL  STGIN                                                  RETAP
      CALL  AGETA                                                  RETAP
      GO TO 10                                                     RETAP
    2 CONTINUE                                                     RETAP
C                                                                  COMN
C       I-B    STORAGE RETRIEVAL  INITIALIZATION  FOR   TRIAL      COMN
C              OR   SOLUTION   TRAJECTORY                          COMN
      CALL  PHIN                                                   FNTG
      CALL  STGIN                                                  FNTG
      CALL BGRA                                                    FNTG
      CALL AGETA                                                   FNTG
      IF(JG(1).LT.11) ALPHA=APHO+ROI                               P014
C                                                                  COMN
C     I-C IF   TRIAL TRAJECTORY  GO TO  I-D ,ELSE  WHEN SIZING FLAG IS COMN
C        SET,  SAVE  INITIAL CONDITION                             COMN
      IF(ITER.NE.3) GO TO 8                                        PHISZ
      IF(LUM.LE.1.AND.JTYP.EQ.0)GO TO 10                           PHISZ
      CALL  INTERI                                                 PHISZ
      GO TO 10                                                     PHISZ
C                                                                  COMN
C       I-D    RETRIEVE  DATA  FOR  FIRST  CONTROL  AND  PARAMETER  CALC. COMN
C              ON  TRIAL   TRAJECTORY                              COMN
    8 CONTINUE                                                     PHISZ
      CALL BGRB                                                    FNTG
      CALL BGET                                                    FNTG
      TIMEPH=TIME-TOP                                              FNTG
      TIMES=TIME-TOS                                               FNTG
      CALL GETIT                                                   FNTG
C               IF ANY PARAMETERS, PERFORM INITIAL STATE CALC.     COMN
C               AND  CALCULATE   CONTROL                           COMN
      IF(NPARA.GT.0) CALL ADJUS                                    FNTG
      CALL MTX                                                     FNTG
      CALL MTX1                                                    FNTG
      JSK=1                                                        FNTG
C                                                                  COMN
C          II       INTEGRATION  ROUTINE                           COMN
   10 CALL  RKUTTA                                                 FNTG
C          II-A     TRAFFIC  FLAG  DIRECTOR                        COMN
     GO TO (20,40,380),L                                           FNTG
C     III  CALCULATE  EQUATIONS OF MOTION  DERIVATIVES             COMN
   20 CALL EQUA                                                    FNTG
      CALL MODELA(ITT,STT)                                         FNTG
C       III-A    COMPUTE  CONTROL  AND  DERIVATIVES                COMN
   30 STT = TIME                                                   FNTG
      ITT = IARC                                                   FNTG
C       III-B    TEST  FOR  CONTROL DIVERGANCE                     COMN
      IF(ISTART.NE.6)                                              FNTG
C       III-C    CHECK  CUT-OFF  REFINEMENT FLAG                   COMN
     1GO  TO (10,370),ICOP                                         FNTG
      CALL OUTERR                                                  P014
```



```
      RETURN                                                      FNTG
C       IV      ARC   CUT-OFF  AND  PHASE  CUT-OFF  TESTING       COMN
C                                                                 COMN
C       IV-A     IS  CUT-OFF  ALREADY IN  PROCESS                 COMN
C                NO - IV-B   , YES - VIII                         COMN
   40  IF(KST NE 0)  GO TO 340                                    FNTG
C                                                                 COMN
C        IV-B    PHASE  CUT-OFF  TESTS                            COMN
C                IF DETECTED  IN LESS  THAN  NEXT INTERVAL        COMN
C                SET  KST =1  AND  COMPUTE  DTP (INTERVAL TO HIT OMP)  COMN
C                ALSO  SET-UP  DATA TO  REFINE  DT  (CALL PSD)    COMN
      IF(JPS-2) 50,60,70                                          FNTG
   50 IF(TIME+DT.LT.OMP-1.E-7) GO TO 100                          FNTG
      DTP = OMP - TIME                                            FNTG.
      GO TO  90                                                   FNTG
   60 IF(TIMEPH+DT.LT.OMP-1.E-7) GOTO 100                         FNTG
      DTP=  OMP - TIMEPH                                          FNTG
      GO  TO  90                                                  FNTG
   70 CALL  PHSTOP (IVR    )                                      FNTG
      IF(IVR.EQ.0) GO TO 100                                      FNTG
      IVAR=ISTPP                                                  FNTG
      IF(JPS.GT 10) CALL SETFPO                                   APR27
   80 CALL PSD( IVAR,OMP)                                         FNTG
      DTP = DTF(IVAR,OMP   )                                      FNTG
   90 KST=1                                                       FNTG
C                                                                 COMN
C       IV-C   ARC  CUT-OFF  TESTS. IF  DETECTED                  COMN
C                 SET   KST =2  AND  COMPUTE  DTS                 COMN
  100 IF(JS -2) 110,120,130                                       FNTG
  110 IF(TIME+DT LT OMG(IARC)-1 E-7) GO TO 220                    FNTG
      DTS = OMG(IARC)- TIME                                       FNTG
      GO TO 150                                                   FNTG
  120 IF( TIMES +DT.LT OMG(IARC) -1 E-7) GO TO 220                FNTG
      DTS =  OMG(IARC) - TIMES                                    FNTG
      GO TO 150                                                   FNTG
  130 CALL  STSTOP(IVR    )                                       FNTG
      IF(IVR.EQ 0) GO TO 220                                      FNTG
C                                                                 COMN
C        IV-D   TEST  FOR   STATE  INEQUALITY  CUT-OFF  SITUATION COMN
C                 WHERE  CUT-OFF  IS  NOT  REACHED (IVAR LESS THAN ZERO)  COMN
C                 ALSO  TEST  FOR  CONCURRENT  PHASING            COMN
C          IF  SITUATION  EXISTS, MAKE  CURRENT  TIME  A  CUT-OFF TIME  COMN
      IF(IVAR.GE.0) GO TO 135                                     APR27
      KST=2                                                       APR27
      IF(OMP EQ OMG(IARC)) KST=3                                  APR27
      GO TO 180                                                   APR27
  135 CONTINUE                                                    APR27
      IVAR=ISTOP                                                  FNTG
  140 CALL  PSD(IVAR,OMG(IARC))                                   FNTG
      DTS = DTF(IVAR,OMG(IARC)  )                                 FNTG
  150 IF(KST NE.1) GO TO 200                                      FNTG
C        IV-E    CONCURRENT  EVENT  TEST                          COMN
      IF(ABS(DTS-DTP).LT.00001) GO TO 190                         FNTG
      IF( DTS - DTP)  200,200,160                                 FNTG
C        IV-F     PHASING  ONLY , CHECK  FOR  PHASE SEQUENCING    COMN
C                  ON  FIRST  NOMINAL  TRAJECTORY                 COMN
  160 IF(IND EQ.1 AND.ICOR(IPST) GT IPH) GO TO 630                FNTG
      DT = DTP                                                    FNTG
      KST=1                                                       FNTG
C        IV-G    SET  EVENT TIME ( STILL APPROXIMATION)           COMN
  170 TIMPR = TIME +DT                                            FNTG
      L=4                                                         FNTG
C        IV-H    TEST  FOR  HAVING PASSED  CUT-OFF                COMN
      IF(DT  GE. 1 E-4) GO TO       250                           SEP18
C         CUT-OFF  TIME PASSED,TREAT CURRENT TIME AS  CORNER      COMN
  180 L=2                                                         FNTG
      DT=DTNC                                                     FNTG
      GO TO (600,230,230),KST                                     FNTG
```

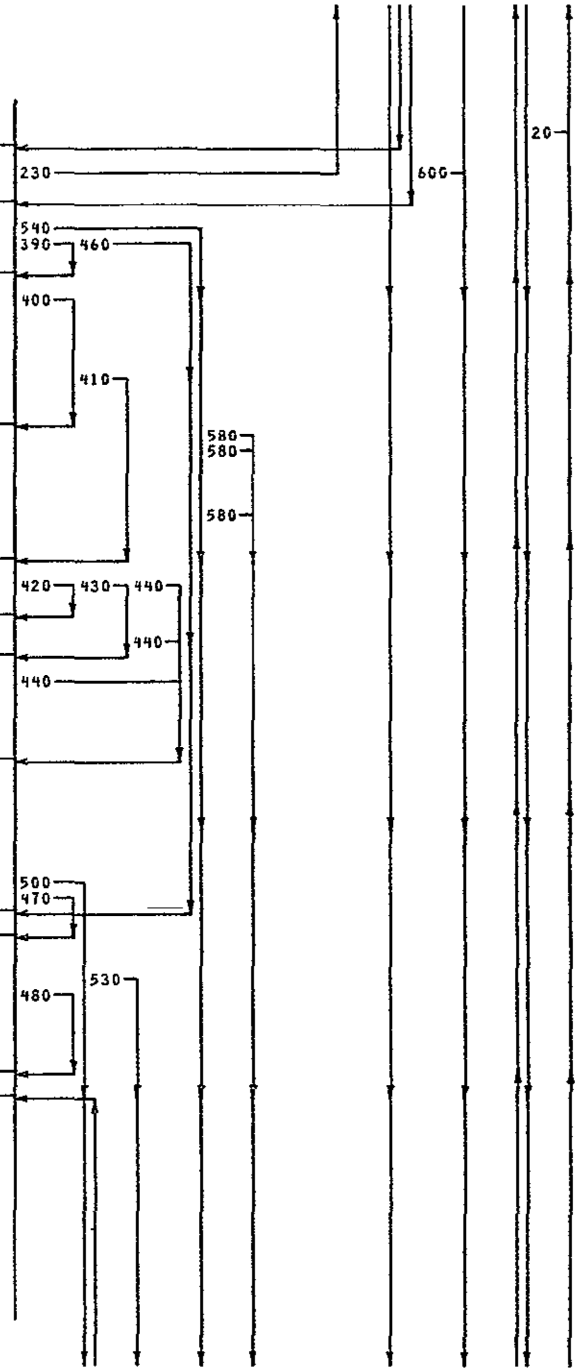


0 OCT 72 G 01-46

```
C        SET  CONCURRENT EVENT STATUS                                  COMN
    190 KST =3                                                         FNTG
        GO TO  210                                                     FNTG
C        SET  ARC-CUT-OFF  EVENT  STATUS                               COMN
    200 KST =2                                                         FNTG
    210 DT= DTS                                                        FNTG
        GO TO 170                                                      FNTG
C        TEST  CUT-OFF FLAG  AFTER ARC-CUT-OFF DETECTED                COMN
    220 IF(KST EQ.1) GO TO 160                                         FNTG
        GO TO  250                                                     FNTG
    230 TIMPR=TIME                                                     FNTG
C        SAVE  TIME  AT ARC TERMINUS AND SAVE CURRENT DERIVATIVES      COMN
C        ALSO  SAVE  TRUNCATED  COMPUTE  INTERVAL                      COMN
        TST(IARC)=TIME                                                 FNTG
    240 CALL SVADYS                                                    FNTG
        DTS(IARC)=DT                                                   FNTG
C         TEST  FOR   CONCURRENT  EVENT                                COMN
        IF(KST EQ 3) GO TO 600                                         FNTG
    250 CONTINUE                                                       FNTG
C                                                                      COMN
C         V   STORE  TRAJ. DATA                                        COMN
        IF(ITER EQ.3) GO TO  330                                       FNTG
        CALL  ASTO                                                     FNTG
    260 CONTINUE                                                       FNTG
C         VI        *   *   *                                          COMN
C         OPTIMAL STAGE TIME  CALCULATION                              COMN
C    VI-A  TEST  FOR  OPTIMAL   STAGING                                COMN
        IF(ITI EQ 0) GO TO 290                                         FNTG
        IF(IOPEN.EQ.2) GO TO 280                                       FNTG
        IF(JSK EQ.0) GO TO 270                                         FNTG
        JSK =0                                                         FNTG
        GO TO 280                                                      FNTG
C       VI-B    CLOSED  LOOP  CONTROL  CALCULATION  OF  TR  VECTOR     COMN
    270 IF(JGII LE.10) CALL GETIT                                      FNTG
        IF(KST EQ 0) CALL MTX                                          FNTG
C       VI-C    CALCULATE  CONTROL  AND  PARAMETER CHANGES             COMN
    280 CALL  MTX1                                                     FNTG
C       VI-C    CALCULATED  NEW  ARC-TIME  CUT-OFF                     COMN
        CALL  ADJUT                                                    FNTG
        GO TO 320                                                      FNTG
    290 CONTINUE                                                       FNTG
C       VII                   TR  LOGIC  WITH NO  OPTIMAL STAGING      COMN
        IF( IND EQ.1.OR.IOPEN.EQ 2) GO TO 320                          FNTG
        IF(JGII          .LE.10) GO TO 320                             FNTG
        IF(JSK EQ.0) GO TO 300                                         FNTG
        JSK = 0                                                        FNTG
        GO TO 310                                                      FNTG
    300 CALL MTX                                                       P014
    310 IF(TIME GT TOPEN) IOPEN =2                                     FNTG
C       VII      SAVE   CURRENT  STATE  AND  START NEXT INTERVAL       COMN
    320 CALL SAVVR                                                     FNTG
        GO TO  10                                                      FNTG
    330 IF(LUM NE 0) CALL ASTOF                                        FNTG
        CALL SAVVR                                                     FNTG
        GO TO 10                                                       FNTG
C       VIII     INTERMEDIATE  CUTTOFF  LOGIC                          COMN
    340 IF(KST GT 3) GO TO 350                                         FNTG
        KST = 4*KST                                                    FNTG
        GO TO 10                                                       FNTG
C         IX     AT  APPROXIMATE  CUT-OFF  TIME                        COMN
    350 KST= KST /4                                                    FNTG
        GO TO (590,360,360),KST                                        FNTG
C        IX-A   IF TIMED ARC, CUT-OFF IS PRECISE  ELSE                 COMN
C               REFINE  STATE AND  TIME  AT  CUT-OFF                   COMN
    360 IF(JS LE 2) GO TO 230                                          FNTG
        IF(TOLF(C).EQ 0 )GO TO 230                                     FNTG
        CALL  YREF( IVAR,OMG(IARC))                                    FNTG
```

OCT 72 G 01-46

```
      ICOP =2                                                        FNTG
      GO TO 20                                                       FNTG
  370 ICOP =1                                                        FNTG
      GO TO (600,230,230),KST                                        FNTG
C      X      AT  CORNER  POINT   OR  PRINT TIME(SOLUTION ONLY)      COMN
  380 IF(ITER NE 2) CALL OUT                                         FNTG
      IF (KST EQ 0)  GO TO 540                                       FNTG
      GO TO (460,390,390),KST                                        FNTG
C      X-A   UPDATE  DATA  AND  INITIALIZE FOR  NEXT ARC             COMN
  390 IARC=IARC+1                                                    FNTG
      IF(IARC GT NARC) GO TO 400                                     FNTG
      CALL READMS(9,ARCDA,51,IARC)                                   FRAT
      DT= DTNC                                                       FNTG
      JS=IABS(JST(IARC))                                             APR27
      CALL PROPIN                                                    FNTG
      GO TO 410                                                      FNTG
C      X-B   AT  END  OF  LAST  ARC   CHECK PHASE SEQUENCING         COMN
C            ON  FIRST NOMINAL  ELSE  END  TRAJECTORY                COMN
  400 IF(IND EQ 2) GO TO 580                                         FNTG
      IF(ICOR(IPST).NE IPH) GO TO 580                                FNTG
      JPH(IPST,2)=JPH(IPH,1)                                         FNTG
      OMGP(IPST,2)= OMGP(IPH,1)                                      FNTG
      CALL MNSET                                                     FNTG
      GO TO 580                                                      FNTG
C      X-C   BRANCHING  LOGIC  AND  INTERMEDIATE  CONSTRAINT TESTS   COMN
C                                                                    COMN
  410 CONTINUE                                                       FNTG
      IF(INTB-1) 440,420,430                                         FNTG
C     CHECK FOR INTERMEDIATE  CONSTRAINT AT ARC  INITIATION          FNTG
  420 IF(IARC-1 EQ NSB) CALL INTBC                                   FNTG
      GO TO 440                                                      FNTG
  430 IF(IARC-1 EQ NSB) CALL BRST                                    FNTG
      IF(IARC-1 NE NSB +NSAB ) GO TO 440                             FNTG
      CALL  BRIN                                                     FNTG
      TOPEN=WORK(2)                                                  FNTG
      IF(TIME LE TOPEN) IOPEN=1                                      FNTG
C      X-D      WEIGHT DISCONTINUITY                                 COMN
  440 CALL  MSDISC                                                   FNTG
  450 CONTINUE                                                       FNTG
      TOS= TIME                                                      FNTG
      IF(IND EQ 1 AND KSOL NE 0) CALL STGIN                          RETAP
      IF(IND EQ 2) CALL STGIN                                        FNTG
      ISTOP=JS-2                                                     FNTG
      ISST = ISIGN( 1, JST(IARC))                                    FNTG
      IF(KST EQ 2) GO TO 500                                         FNTG
      GO TO 470                                                      FNTG
  460 KPST =1                                                        FNTG
  470 CONTINUE                                                       FNTG
C      X-C      PHASE  CORNER POINT  AND  NEXT PHASE  INITIATION     COMN
      IF(IND EQ 2) GO TO 530                                         FNTG
      IF(ICOR(IPST) GT IPH  )  GO TO 480                             FNTG
      JPH(IPST,2)  = JPH(IPH,1)                                      FNTG
      OMGP(IPST,2) = OMGP(IPH,1)                                     FNTG
      CALL MNSET                                                     FNTG
      IPST = IPST +1                                                 FNTG
  480 IPH =IPH+1                                                     FNTG
  490 TOP= TIME                                                      FNTG
      JPS= IABS(JPH(IPH,IND))                                        FNTG
      CP=1.                                                          FNTG
      ISTPP=JPS-2                                                    FNTG
      IF(ISTPP EQ 2) CP=RDI                                          FNTG
      ISPH= ISIGN(1,JPH(IPH,IND) )                                   FNTG
      JGII=JGIO(IPH,IND)                                             FNTG
      OMP = OMGP(IPH,IND)                                            FNTG
      IF(IND EQ 2) CALL PHIN                                         FNTG
      IF(IND.EQ 1 AND KSOL NE.0) CALL PHIN                           RETAP
      DT=DTNC                                                        FNTG
C                                                                    COMN
C      X-D  CONTINUE  NEXT ARC INITIALIZATION  AND  SET              COMN
```



```
C          UP  TR VECTOR  FOR  NEXT ARC  IF  STILL  CLOSED LOOP        COMN
    500 KST =0                                                         FNTG
        TIMPR =TIME                                                    FNTG
        L=4                                                            FNTG
        IF(ITER EQ.3 OR.IND EQ 1) GO TO 10                             FNTG
        IF(NPA  EQ.0) GO TO 510                                        FNTG
        CALL ADJIN                                                     FNTG
        IF(ITI EQ.0) GO TO 510                                         FNTG
        IF(IOPEN EQ 2)  GO TO 10                                       FNTG
        GO TO  520                                                     FNTG
    510 IF(IOPEN.EQ.2) GO TO 10                                        FNTG
        IF(JGII.LE 10) GO TO 10                                        FNTG
    520 CONTINUE                                                       FNTG
        TIMES=TIME-TOS                                                 FNTG
        TIMEPH=TIME-TOP                                                FNTG
        CALL GETIT                                                     FNTG
        CALL MTX                                                       FNTG
        JSK =1                                                         FNTG
        GO TO 10                                                       FNTG
C                                                                      COMN
C        X-E     NEXT PHASE  INITIALIZATION                            COMN
    530 IPH = IPH +1                                                   FNTG
        IPST = IPH                                                     FNTG
        GO TO 490                                                      FNTG
    540 IF(ITER NE 3)GO TO 560                                         FNTG
        IF(KPST EQ.1) GO TO 550                                        FNTG
        TIMPR= TIMPR + DTPI*DTMC                                       FNTG
        GO TO  10                                                      FNTG
C        XI      NEW  PRINT  TIME   UPDATE                             COMN
    550 KPST =0                                                        FNTG
        TIMPR= TIMPR + DTPI*DTMC                                       FNTG
        GO TO  10                                                      FNTG
    560 IF(KPST NE.1) GO TO 570                                        FNTG
    570 TIMPR = TIME + 1.E6                                            FNTG
        KPST =0                                                        FNTG
        GO  TO  10                                                     FNTG
C                                                                      COMN
C        XII   FINAL  TRAJECTORY  WRAPUP                               COMN
    580 CONTINUE                                                       FNTG
        NPH=IPST                                                       FNTG
        TPH(IPST)=TIME                                                 FNTG
        NST=IARC-1                                                     FNTG
        IF(ITER EQ 2) CALL OUT                                         FNTG
        IF(ITER.NE.3 AND IPFLG1.EQ.0) CALL PRORPA                      P014
        CALL ENDWR                                                     FNTG
        IF(JTYP.GT.0) CALL TRJEND                                      PH15Z
        RETURN                                                         FNTG
    590 IF(JPS LE.2) GO TO 600                                         FNTG
C       IX-B     PHASE TIME AND STATE  REFINEMENT                      COMN
        IF(TOLPH(C).EQ.0.) GO TO 600                                   FNTG
        CALL  YREF( IVAR,DMP)                                          FNTG
        ICOP =2                                                        FNTG
        GO TO 20                                                       FNTG
    600 IF(IND EQ 2) GO TO  610                                        FNTG
C       IX-C     PHASE TIME IS PRECISE . CHECK PHASE  SEQUENCING       COMN
C                   ON  FIRST NOMINAL TRAJ                             COMN
        IF(ICOR(IPST).GT.IPH)GO TO 620                                 FNTG
    610 TPH(IPST)= TIME                                                FNTG
        TIMPR=TIME                                                     FNTG
        DIP(IPST)= DT                                                  FNTG
        C
        GO TO  250                                                     FNTG
    620 TIMPR = TIME                                                   FNTG
        GO TO  250                                                     FNTG
    630 TIMPR=TIME                                                     FNTG
        KST=1                                                          FNTG
        GO TO 10                                                       FNTG
        END                                                            FNTG
```

SUBRØUTINE FXDAT

Subroutine FXDAT

Purpose

Subroutine FXDAT fixes floating point numbers in the /ARCDAT/ common block, establishes some presets, and prints the data contained in /ARCDAT/.

Description

In addition to fixing input option flags and curve numbers, FXDAT also establishes presets if certain critical data has not been input. After this, it prints out the arc data.

FXDAT is called from subroutine GEINP for each set of arc data. The single argument, N is the arc number.

474

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ARCDA	S_{ref}	I	Aerodynamic reference area (FT^2)	/ARCDAT/(	1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SDINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF
FRATE		I	Input rated vacuum thrust per engine (LBS)	/ARCDAT/(	42)	EQUA3	I	FRATE
						FXDAT	I	FRATE
						PROPB	I	FRATE
						PROPIN	I	FRATE
						SIZIN	O	FRATE
IARCDA	S_{ref}	O	Aerodynamic reference area (FT^2)	/ARCDAT/(	1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SDINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF
IATM		I	Atmosphere option flag	/ARCDAT/(	7)	EQUA3	I	IATM
						FXDAT	I	IATM
						OUT	I	IATM
						PDBC	I	IATM
						VT	I	IATM
IMODE		M	Control mode option flag	/ARCDAT/(	8)	FXDAT	M	IMODE
						SDINP	I	IMODE
MT		I	Curve number -thrust table	/ARCDAT/(	25)	FXDAT	I	MT
						PROPB	I	MT
						PROPIN	I	MT
						THRUST	I	MT
N		I	Arc number	/FXDAT /(*	)	FXDAT	I	N
QMULT	=0 OR 1	O	Heating flag multiplier	/ARCDAT/(	40)	DER3A	I	QMULT
						FXDAT	O	QMULT
						PDY3A	I	QMULT
RHOB	ρ_b	O	Atmosphere base density for heating calculation (LB/FT**3)	/ARCDAT/(	39)	DER3A	I	RHOB
						FXDAT	O	RHOB
						PDY3A	I	RHOB
TMULT	T_{mult}	M	Thrust multiplier or number of engines	/ARCDAT/(	4)	EQUA3	I	TMULT
						FXDAT	M	TMULT
						PROPB	I	TMULT
						PROPIN	I	TMULT
						SIZIN	O	TMULT

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
UN06.		0	File of all output data	/.UN06./(\$	)	BLICO	0	UN06.
						BNDRYC	0	UN06.
						CRASH	0	UN06.
						FRENCH	0	.UN06.
						FXDAT	0	.UN06.
						GEINP	0	UN06.
						HUNT	0	.UN06.
						INEDIT	0	.UN06.
						ITER8	0	UN06.
						MODELA	0	UN06.
						MOMJ	0	UN06.
						MPSI	0	UN06.
						OUT	0	UN06.
						PAYO2	0	UN06.
						PRINT	0	.UN06.
						PRINTV	0	.UN06.
						PRINTW	0	.UN06.
						PRITEQ	0	.UN06.
						PRITVA	0	.UN06.
						PROPIN	0	UN06.
						PROTHR	0	.UN06.
						PRWTSM	0	.UN06.
						RANGE	0	.UN06.
						S	0	.UN06
						SDINP	0	UN06.
						SIZE	0	.UN06.
						SIZIN	0	UN06.
						SIZOUT	0	.UN06.
						SOLVE	0	.UN06.
						SPLICO	0	.UN06
						SPLIZ	0	.UN06.
						SPLYNE	0	.UN06.
						SSSP	0	.UN06.
						STAU	0	.UN06.
						STPIT	0	.UN06.
						SUMOUT	0	.UN06.
						TABIN	0	.UN06.
						TEST	0	UN06.
						VEHDF	0	.UN06.
						WTSCH	0	.UN06.
						WTVOL	0	.UN06.

```
      SUBROUTINE FXDAT(N)                                               FXDAT
C                                                                       COMN
C        FIXES  FLOATING  POINT NUMBERS  IN  ARCDAT (INPUT DATA)        COMN
C        AND   ESTABLISHES  SOME  PRESETS  ALSO  PRINTS  DATA           COMN
C                                                                       COMN
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTP1      , ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      , ARCDAT
     *XLMAX     ,HOMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      , ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      , ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      , ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        , ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                  ARCDAT
     *  ,FRATE  ,  ARCD(9)                                              RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      DIMENSION  IARCDA(1)                                              FXDAT
      EQUIVALENCE (IARCDA,ARCDA)                                        FXDAT
      DO 10 I=7,10                                                      FXDAT
   10 IARCDA(I)=ARCDA(I)                                                FXDAT
      DO 20 I=18,31                                                     FXDAT
   20 IARCDA(I)= ARCDA(I)                                               FXDAT
      IARCDA(38)=ARCDA(38)                                              FXDAT
      IF(IATM-1) 30,40,50                                               FXDAT
C         62  STANDARD                                                  FXDAT
   30 RHOB = 2 3769054E-3                                               FXDAT
      GO TO 60                                                          FXDAT
C         63 PAT                                                        FXDAT
   40 RHOB = 2 2964063E-3                                               FXDAT
      GO TO 60                                                          FXDAT
   50 QMULT=0.                                                          FXDAT
      RHOB =1.                                                          FXDAT
   60 CONTINUE                                                          FXDAT
      IF(TMULT GT.0.) GO TO 65                                          FRAT
      IF(FRATE GT.0.OR.MT NE.0) TMULT =1                                FRAT
   65 CONTINUE                                                          FRAT
      IF(IMODE.EQ.0) IMODE=1                                            FRAT
      WRITE(6,70) N,(ARCDA(I),I=1,17),(ARCDA(J),J=32,37) ,ARCDA(41),    FXDAT
     *ARCDA(42),                                                        FRAT
     1(ARCDA(K),                                                        FXDAT
     2K=18,31),ARCDA(38)                                                FXDAT
      RETURN                                                            FXDAT
   70 FORMAT(//17H  DATA FOR SUBARCI3 / 5X,5HSREF=E12.6,4X,9HNOZ.AREA=E1 FXDAT
     12 6,9X,4HISP=F7 3,9X,9HTH  MULT=F5 2,12X, 8HDELTA T=F9.5/         FXDAT
     210H PRT MULT=F5.1,11X,9HATM OPTN=I2,14X,9HCNT.MODE=I3,13X,9HAER OP FXDAT
     *TN=I3,13X,9HPRO OPTN=I3/ 5X,5HQMAX=F7 2,13X,5HGMAX=F6 3,          APR27
     410X,9HMAX LIFT=E12.6,                                             FXDAT
     5                                                        5X,8HHEAT FXDAT
     6 RT=E12.6,7X,6HGMDOT=E13.6/3X,7HALFMAX=F7.3,11X,7HPHIMAX=,F7.3    P014
     7,11X, 7HXCGREF=F8.3,10X, 7HZCGREF=F7 3,13X, 5HXENG=F8.3/          FXDAT
     * 5X, 5HZENG=F7.3,12X,6HXTAIL=F8.3,12X,5HDREF=F7.3,11X,6HREMAX=    P014
     * E12.4,11X,6HFRATE=E12.4/                                         NOS
     X15H  TABLE NUMBERS/4X,6HAERO AI4,7H AERO BI4,7H AERO CI4,7H AERO D FXDAT
     AI4,7H AERO EI4,7H AERO FI4,7H AERO GI4,7H THRUSTI4,7H ISP LSI4,   FXDAT
     B7H    XCGI4,7H    ZCGI4/ 4X,6HWIND AI4,7H WIND BI4                FXDAT
     C,7H BASE DI4,7H    CNDI4)                                         FXDAT
       END                                                              FXDAT
```

SUBRØUTINE GETIT

Subroutine GETIT

Entry Points. STGIN, PHIN

Purpose

GETIT controls trajectory and adjoint data fetching during forward trial trajectory integration.

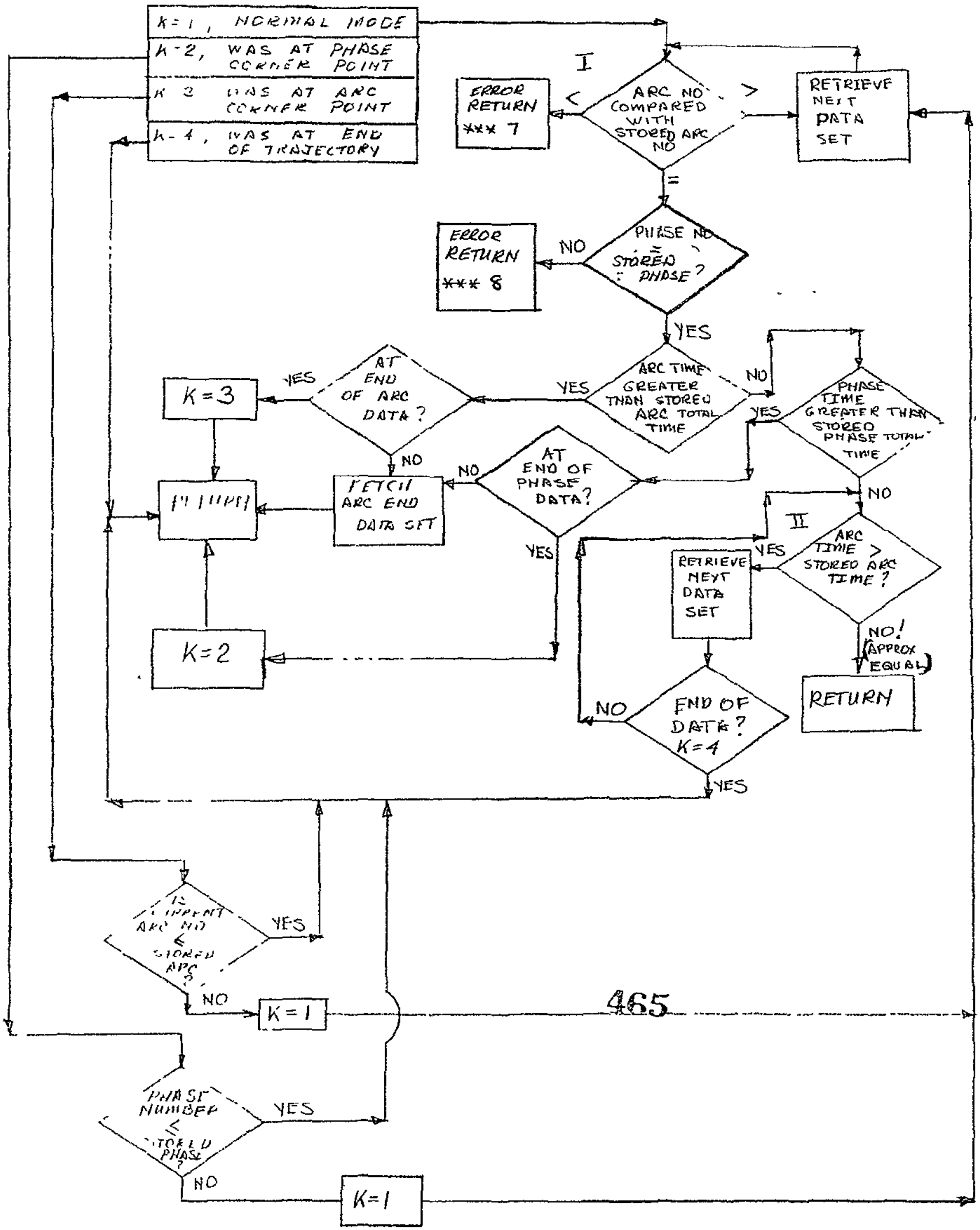
Description

GETIT retrieves data by calling subroutines AGETA and BGET. The main intent of the logic in this routine is to retrieve data from the adjoint and old nominal trajectory at as near the current arc time as possible. (Arc time rather than elapsed time is the independant variable in the program). No data interpolation is done.

Entry points STGIN and PHIN are called by FNTG at the beginning of each arc and phase respectively to locally define total arc and phase durations.

GETIT is called by MODELA.

SUBROUTINE GETIT



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DDT		W	Difference between stored and current arc time	/GETIT /(*	)	GETIT	W	DDT
DIP1		I	Phase initial times for nominal trajectory [sd]	/GENF /(	453)	GETIT	I	DIP1
						SDINP	M	DIP1
						TEST	O	DIP1
						TOPM	I	DIP1
DIS1		I	Arc initial times for nominal trajectory [sd]	/GENF /(	473)	GETIT	I	DIS1
						SDINP	M	DIS1
						TEST	O	DIS1
						TOPM	I	DIS1
						TRAN3	I	DIS1
DPHT		W	Stored phase duration	/GETIT /(*	)	GETIT	W	DPHT
DST	τ_{OLD}	W	Stored arc duration	/GETIT /(*	)	GETIT	W	DST
FTIME		I	Time at which trajectory data set is stored. (SEC)	/RETREV/(	1)	AGETB3	O	FTIME
						AST3	O	FTIME
						GETIT	I	FTIME
						TRAN3	I	FTIME
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
IPH		I	Phase number	/XCODES/(	143)	ADID3A	I	IPH
						ADJUST	I	IPH
						AST3	I	IPH
						BNTG	M	IPH
						FNTG	M	IPH
						GETIT	I	IPH
						GUI3A	I	IPH
						SDINP	M	IPH
IPHN		I	Stored history data phase number	/XCODES/(	170)	AGETB3	M	IPHN
						AST3	M	IPHN
						GETIT	I	IPHN
ISTN		I	Stored history data arc number	/XCODES/(	169)	AGETB3	O	ISTN
						AST3	O	ISTN
						GETIT	I	ISTN
						TRAN3	I	ISTN
ITER		I	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
						FNTG	I	ITER
						GETIT	I	ITER
						MODELA	I	ITER
						OUT	I	ITER
						PAYO2	M	ITER
						PROPIN	I	ITER
						TEST	M	ITER
						TOPM	M	ITER
K		M	Storage retrieval flag indicates end of arc, phase, or data.	/XCODES/(	156)	AST3	O	K
						FNTG	O	K
						GETIT	M	K
						MODELA	I	K
						SDINP	M	K
TIMEPH	τ_P	I	Phase time (SEC)	/GENF /(	318)	EQUA3	O	TIMEPH
						FNTG	M	TIMEPH
						GETIT	I	TIMEPH
						GUI3A	I	TIMEPH
						OUT	I	TIMEPH

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
TIMES	τ	I	Arc time	(SEC)	/GENF	/(319)	AST3	I	TIMES
							EQUA3	O	TIMES
							FNTG	M	TIMES
							GETIT	I	TIMES
							OUT	I	TIMES
TMS		W	Stored current arc time		/GETIT	/(*)	GETIT	W	TMS
TPH1		I	Phase end times for nominal trajectory		/GENF	/(413)	BNTG	I	TPH1
							GETIT	I	TPH1
							SDINP	O	TPH1
							TEST	O	TPH1
							TOPM	I	TPH1
TST1		I	Arc end times for nominal trajectory		/GENF	/(433)	BNTG	I	TST1
							GETIT	I	TST1
							PROPIN	I	TST1
							SDINP	O	TST1
							TEST	O	TST1
							TOPM	I	TST1
							TRAN3	I	TST1
							TRTOSZ	I	TST1

```
      SUBROUTINE GETIT                                                    GETIT
C                                                                         COMN
C          CONTROLS  STORAGE FETCHING FOR FORWARD  TRAJECTORY             COMN
C          K  IS THE  KEY  FLAG                                           COMN
C             K=1   NORMAL  PROGRESSION                                   COMN
C             K=2   HUNG  UP  AT    PHASE END POINT                       COMN
C             K=3   HUNG  UP  AT    ARC   END POINT                       COMN
C             K=4   AT  END  OF  STORED  DATA                             COMN
C                                                                         COMN
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                             RETREV
      COMMON/RETREV/                                                      RETREV
     *NBUFA(2)  ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA      ,  RETREV
     *MIXB      ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB        RETREV
     *,NBUFB    ,IBUFB                                                    RETREV
      COMMON /XCODES/                                                     XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2),  XCODES
     *JST   (20)          ,NCNST     ,NSB       ,NSAB      ,NICNB     ,  XCODES
     *I20P      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       ,  XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    ,  XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS        ,  XCODES
     *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE     ,  XCODES
     *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         ,  XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     ,  XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         ,  XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB      ,  XCODES
     *NCTIN     ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/GENF/                                                        GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,UDC(20)   ,  GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,OCON(9)   ,DTP       ,  GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS        ,  GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS        ,  GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     ,  GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     ,  GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         ,  GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP       ,  GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,  GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     ,  GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                        GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     ,  GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      ,  GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA     ,  GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,  GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       ,  GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                      GENF
      COMMON / GENF  /                                                    GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       ,  GENF
     *FRATED    ,IRATED    ,                                              GENF
     *P1        ,P2        ,P3       ,XK1        ,XK2       ,XK3       ,  GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,  GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,  GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,  GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1Q      ,XK2Q      ,XK3Q      ,  GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,  GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)    GENF
      REAL  LIFTR  , LIFT  , LIFTA,LIFTM  ,              MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV  , IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                         GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                  GENF
      GO  TO (10,100,110,120),K                                           GETIT
C       I      LOOP   IF  ARC NO  GT.  STORED  ARC  NUMBER                COMN
   10 IF( IARC- ISTN) 140,30,20                                           GETIT
   20 CONTINUE                                                            GETIT
      CALL AGETA                                                          GETIT
      IF (ITER NE 3) CALL BGET                                            GETIT
      GO  TO  10                                                          GETIT
   30 IF( IPH NE IPHN) GO TO 150                                          GETIT
      IF ( TIMES.GT. DST) GO TO 50                                        GETIT
      IF ( TIMEPH.GT DPHT) GO TO80                                        GETIT
C       II      LOOP    IF    CURRENT ARC TIME   GT   STORE ARC TIME      COMN
   40 TMS=FTIME-TST1(IARC)                                                GETIT
      DDT=TIMES-TMS                                                       GETIT
      IF(ABS(DDT) LE.1.E-3) GO TO 130                                     GETIT
      IF(DDT.LT 1.E-5) GO TO 130                                          GETIT
```

20 OCT 72 G 01-46

```
      CALL  AGETA                                                       GETIT
      IF(ITER NE 3) CALL BGET                                           GETIT
C        II-A   STOP  SEARCH  IF  END OF  STORED DATA                   COMN
      IF(K EQ.4) GO TO 130                                              GETIT
      GO TO 40                                                          GETIT
   50 IF(TMS GE DST) GO TO 70                                           GETIT
C        II-B    STOP  LOOP    IF  ARC TIME  GREATER THAN STORED ARC    COMN
C                  DURATION   (SET  K=3)                                COMN
   60 CALL AGETA                                                        GETIT
      IF(ITER NE.3) CALL BGET                                           GETIT
      TMS=FTIME-TST1(IARC)                                              GETIT
      GO TO 130                                                         GETIT
   70 K=3                                                               GETIT
      GO TO 130                                                         GETIT
C         III    STOP  IF   PHASE TIME  GT STORED PHASE DURATION        COMN
   80 TMP=FTIME-TPH1(IPH)                                               GETIT
       IF(TMP.GE DPHT) GO TO 90                                         GETIT
      GO TO 60                                                          GETIT
   90 K=2                                                               GETIT
      GO TO 130                                                         GETIT
  100 IF(IPH LE IPHN) GO TO 130                                         GETIT
      K =1                                                              GETIT
      GO TO 20                                                          GETIT
  110 IF(IARC.LE ISTN) GO TO 130                                        GETIT
      K=1                                                               GETIT
      GO TO  20                                                         GETIT
  120 CONTINUE                                                          GETIT
  130 RETURN                                                            GETIT
C        IV                                                             COMN
C        ENTRY POINTS  SET-UP  ARC AND PHASE DURATIONS                  COMN
C        FROM  STORED  DATA  TO  CORRESPOND TO  CURRENT                 COMN
C        ARC  AND  PHASE                                                COMN
      ENTRY STGIN                                                       GETIT
      DST = DIS1(IARC) - TST1(IARC)                                     GETIT
      RETURN                                                            GETIT
      ENTRY  PHIN                                                       GETIT
      DPHT = DIP1(IPH)-  TPH1(IPH)                                      GETIT
      RETURN                                                            GETIT
  140 CONTINUE                                                          GETIT
      CALL OUT                                                          GETIT
      IF(ICOP NE 2) CALL STPIT(7)                                       GETIT
      GO TO 160                                                         GETIT
  150 CALL STPIT(8)                                                     GETIT
  160 RETURN                                                            GETIT
      END                                                               GETIT
```

20 OCT 72 6 01-46

SUBRØUTINE
GUI3A

Subroutine GUI3A

Entry Point. GUID

Purpose

GUI3A computes simple nonoptimal control for steepest descent starting solution.

Description

The local option flag JGII determines which control calculation is used. This routine is called from MODELA.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
APHR	α	M	Angle of attack	(DEG)	/AEC03 /(	2)	AGETB3	O	APHR
							AST3	M	APHR
							BEROCO	I	APHR
							BLGCON	O	APHR
							GUI3A	M	APHR
							MODELA	M	APHR
							MODELB	I	APHR
							MTX3A	O	APHR
							OUT	I	APHR
COSPHI	$\cos\phi$	M	See symbol		/AEC03 /(	13)	ACCEL	I	COSPHI
							BL4	I	COSPHI
							GUI3A	M	COSPHI
							MODELA	M	COSPHI
							MODELB	M	COSPHI
							OUT	I	COSPHI
GAM	γ	I	Relative flight path angle	(RAD)	/STATE3/(	2)	EQUA3	I	GAM
							GUI3A	I	GAM
							OUT	I	GAM
IND		I	Flag indicates whether on first nominal trajectory (IND=1)		/XCODES/(	141)	AST3	I	IND
							BGET3	I	IND
							FNTG	M	IND
							GUI3A	I	IND
							MTX3A	I	IND
							PROPIN	I	IND
IPH		I	Phase number		/XCODES/(	143)	ADID3A	I	IPH
							ADJUST	I	IPH
							AST3	I	IPH
							BNTG	M	IPH
							FNTG	M	IPH
							GETIT	I	IPH
							GUI3A	I	IPH
							SDINP	M	IPH
IPST		I	Phase counter for first nominal trajectory		/XCODES/(	167)	AST3	I	IPST
							FNTG	M	IPST
							GUI3A	I	IPST
							REU3	I	IPST
JGII		I	Control option		/XCODES/(	195)	ACCEL	I	JGII
							BNTG	O	JGII
							DER3A	I	JGII
							FNTG	M	JGII
							GUI3A	I	JGII
							MODELA	I	JGII
							MODELB	I	JGII
							MTX3A	I	JGII
							PDY3A	I	JGII
MNGA		M	Control history curve number		/STS /(	32)	GUI3A	M	MNGA
							SDINP	M	MNGA
							SDINP	I	TAL
MNGP		M	Control history curve number		/STS /(	72)	GUI3A	M	MNGP
							SDINP	M	MNGP
							SDINP	I	WTPI
PHID	ϕ	M	Bank angle	(DEG)	/AEC03 /(	10)	AGETB3	O	PHID
							AST3	M	PHID
							GUI3A	M	PHID
							MODELA	M	PHID
							MODELB	I	PHID
							MTX3A	O	PHID
							OUT	I	PHID
PHIR	ϕ	M	Bank angle	(RAD)	/AEC03 /(	11)	GUI3A	M	PHIR
							MODELA	M	PHI
							MODELB	M	PHI
							OUT	I	PHI
PSI	ψ	I	Azimuth		/STATE3/(	5)	EQUA3	I	PSI
							GUI3A	I	PSI
							OUT	I	PSI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
RAD		I	Radian to angle conversion, 57.29577951		/DATA	/(2)	BEROCO	I	DEG
							BLGCON	I	RAD
							ENVPRM	I	RAD
							EQUA3	I	RAD
							FNTG	I	RAD
							GUI3A	I	RAD
							MODELA	I	RAD
							MTX3A	I	RAD
							OUT	I	RAD
							PADS1	O	RAD
							SDINP	I	RAD
							TRTO5Z	I	RAD
RDI		I	Angle to radian conversion, .01745329252		/DATA	/(3)	BLICO	I	RDI
							DER3A	I	RDI
							FNTG	I	RDI
							GUI3A	I	RDI
							MODELA	I	RDI
							MODELB	I	RDI
							PADS1	O	RDI
							PROPB	I	RDI
							PROPIN	I	RDI
							REU3	I	RDI
							SDINP	I	RDI
							SOMG	I	RDI
SINA	$\sin\alpha$	M	See symbol		/AEC03	/(7)	ACCEL	I	SINA
							BL4	I	SINA
							BL6	I	SINA
							BL7	I	SINA
							BL8	I	SINA
							FH3	I	SINA
							GUI3A	M	SINA
							OUT	I	SINA
							VT	M	SINA
THL		I	Steering azimuth attitude.	(DEG)	/GUI3A	/(*)	GUI3A	I	THL
THLR	θ_l	W	Steering azimuth attitude.	(RAD)	/GUI3A	/(*)	GUI3A	W	THLR
THP		I	Steering pitch attitude.	(DEG)	/GUI3A	/(*)	GUI3A	I	THP
THPR	θ_p	W	Steering pitch attitude.	(RAD)	/GUI3A	/(*)	GUI3A	W	THPR
TIMEPH	τ_p	I	Phase time	(SEC)	/GENF	/(318)	EQUA3	O	TIMEPH
							FNTG	M	TIMEPH
							GETIT	I	TIMEPH
							GUI3A	I	TIMEPH
							OUT	I	TIMEPH

```
      SUBROUTINE GUI3A                                                      GUI3A
C                                                                           COMN
C          COMPUTES  SIMPLE NON-OPTIMAL  CONTROL                            COMN
      COMMON/GENF/                                                          GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WOC(20)   ,   GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       ,   GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        ,   GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS        ,   GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     ,   GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     ,   GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         ,   GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       ,   GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,   GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     ,   GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                         GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     ,   GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      ,   GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     ,   GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,   GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       ,   GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                       GENF
      COMMON / GENF  /                                                      GENF
     *XJV        ,XJR       ,GN        ,GAMMAD    ,XKG       ,XKP       ,   GENF
     *FRATED    ,IRATED     ,                                               GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3       ,   GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,   GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,   GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,   GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      ,   GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,   GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)     GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                      FRAT
      DIMENSION TPH1(10),TST1(10)                                           GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                    GENF
      COMMON /XCODES/                                                       XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2),   XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB     ,   XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       ,   XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    ,   XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS        ,   XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE     ,   XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         ,   XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     ,   XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         ,   XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB      ,   XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON/ARCDAT/                                                        ARCDAT
     *SREF       ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      ,   ARCDAT
     *IATM       ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      ,   ARCDAT
     *XLMAX      ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      ,   ARCDAT
     *MAEB       ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      ,   ARCDAT
     *MT         ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      ,   ARCDAT
     *MDB        ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        ,   ARCDAT
     *DREF       ,MCND      ,RHOB      ,QMULT     ,REMAX                    ARCDAT
     * ,FRATE   , ARCD(9)                                                   RETAP
      DIMENSION  ARCDA(40)                                                  ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                               ARCDAT
      COMMON/DATA/                                                          DATA
     *PI         ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      ,   DATA
     *FTNM       ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4          DATA
      COMMON/STS/                                                           STS
     *OPAY       ,PMIN      ,WORK (20),NWDS       ,IPC    (7),NITER     ,   STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)               STS
      COMMON/GLOBAL/                                                        GLOBAL
     *GR         ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM           GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC       ,NBRAN    ,NFARC     ,ID(4)         GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                                  GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)         GLOBAL
     *,ITPSO ,KSOL      ,KGLOBL(8)                                          RETAP
      COMMON/STATE3/                                                        STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)   ,   STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) ,   STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    ,   STATE3D
     *SVBV    (9),OMEGA     ,OMEGA2    ,                                    STATE3D
```

```
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP       ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO       ,UDO       ,HTDV      ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                 STATE3D
      COMMON/STATE3/                                                      STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                        STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,   EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2),  EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),  EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)             EQUV3
      REAL M,MU,MD                                                        EQUV3
      COMMON/AECO3/                                                       AECO3
     *APHD      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       , AECO3
     *SINA      ,COSA      ,PHID      ,PHID      ,PHI       ,SINPHI    , AECO3
     *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       , AECO3
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      , AECO3
     *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       , AECO3
     *CLAM      ,CL        ,CLA       ,CLM       ,                        AECO3
     *CD        ,CDA       ,CDM                                           AECO3
      EQUIVALENCE (PHIR,PHI)                                              IO
      ENTRY GUID                                                          IO
      GO TO (10,20,90),JGII                                               GUI3A
C       I     ARHR  AND  PHID  FUNCTIONS OF  PHASE  TIME                  COMN
   10 CALL TBLK(MNGA(IPH,IND),TIMEPH,APHR)                                GUI3A
      CALL TBLK(MNGP(IPH,IND),TIMEPH,PHID)                                GUI3A
      RETURN                                                              GUI3A
C                                                                         COMN
C     II     PITCH AND YAW  STEERING  ATTITUDES                           COMN
   20 CALL TBLK(MNGA(IPH,IND),TIMEPH,THP)                                 GUI3A
      CALL TBLK(MNGP(IPH,IND),TIMEPH,THL)                                 GUI3A
      THPR =THP*RDI                                                       GUI3A
      THLR =THL*RDI                                                       GUI3A
      ALL = THLR - PSI                                                    GUI3A
      ALP = THPR - GAM                                                    GUI3A
      IF(ALL.EQ  0.) GO TO 60                                             GUI3A
      IF(ALP EQ  0.) GO TO 70                                             GUI3A
      SALL = SIN(ALL)                                                     GUI3A
      SALP = SIN(ALP)                                                     GUI3A
      PHIR= ATAN2( SALL,SALP)                                             GUI3A
      COSPHI=COS(PHIR)                                                    GUI3A
      SINA= SALP/COSPHI                                                   GUI3A
      IF(ABS(SINA).GT.1.) SINA=.999999999                                 GUI3A
      APHR = ABS(ASIN(SINA)) *RAD                                         GUI3A
      PHID= PHIR * RAD                                                    GUI3A
      IF(PHID GT.90.) GO TO 30                                            GUI3A
      IF(PHID LT.-90.) GO TO 40                                           GUI3A
      RETURN                                                              GUI3A
   30 PHID=PHID-180.                                                      GUI3A
       GO TO 50                                                           GUI3A
   40 PHID=180.+PHID                                                      GUI3A
   50 APHR=-APHR                                                          GUI3A
      RETURN                                                              GUI3A
   60 CONTINUE                                                            GUI3A
      PHID= 0.                                                            GUI3A
      APHR = ALP*RAD                                                      GUI3A
      RETURN                                                              GUI3A
   70 IF(ALL LT.0.) GO TO 80                                              GUI3A
      APHR = ALL*RAD                                                      GUI3A
      PHID= 90.                                                           GUI3A
      RETURN                                                              GUI3A
   80 APHR =-ALL*RAD                                                      GUI3A
      PHID=-90.                                                           GUI3A
      RETURN                                                              GUI3A
C                                                                         COMN
C     II     PITCH  STEERING  WITH  BANK =0                               COMN
   90 CONTINUE                                                            GUI3A
      CALL TBLK(MNGA(IPH,IND),TIMEPH,THP)                                 GUI3A
      GAMD=GAM*RAD                                                        GUI3A
```

```
      PHID=0.                                    GUI3A
      APHR=THP-GAMD                              GUI3A
      RETURN                                     GUI3A
      ENTRY MNSET                                GUI3A
      MNGP(IPST,2) =MNGP(IPH,1)                  GUI3A
      MNGA(IPST,2) =MNGA(IPH,1)                  GUI3A
      RETURN                                     GUI3A
      END                                        GUI3A
```

```
      PHID=0.                                                           GUI3A
      APHR=THP-GAMD                                                     GUI3A
      RETURN                                                            GUI3A
      ENTRY MNSET                                                       GUI3A
      MNGP(IPST,2) =MNGP(IPH,1)                                         GUI3A
      MNGA(IPST,2) =MNGA(IPH,1)                                         GUI3A
      RETURN                                                            GUI3A
      END                                                               GUI3A
```

20 OCT 72 6.01-46

FUNCTION

ICOD

FUNCTION ICØD

Entry IVØD

Purpose

The main entry ICOD unpacks the initial condition option code from the packed data boundary condition array word.

The second entry IVOD unpacks the variable code from the word.

Description

As described in GEINP, the input boundary conditions are processed and stored on random file 9 record 21. Subroutine SDINP then reads file 9 and employs ICOD and IVOD to get information from the packed words. This information is used to set up internal codes and check input data.

```
      FUNCTION ICOD(REZ)                                          ICOD
C                                                                 COMN
C      UNPACKS   OPTION CODE  AND  VARIABLE CODE FROM             COMN
C      INITIAL   CONDITION ARRAY                                  COMN
      I= REZ*1.E-6 +SIGN(.5,REZ)                                  ICOD
      Z= I*1.E6                                                   ICOD
      F= REZ-Z                                                    ICOD
      ICOD= F +SIGN( 5,REZ)                                       ICOD
      RETURN                                                      ICOD
      ENTRY  IVOD                                                 ICOD
      ICOD = REZ*1.E-6 +SIGN(.5,REZ)                              ICOD
      RETURN                                                      ICOD
      END                                                         ICOD
```

SUBRØUTINE
IMPUL

Subroutine IMPUL

Purpose

IMPUL computes specific impulse for rocket propulsion options.

Description

This routine has two options, depending on whether JPRO equals 0 or 1. When JPRO = 0, it will compute a corrected ISP if the ISP loss table has been input. When JPRØ = 1, it will always compute a corrected ISP for the dual engine simulation (used with SSSP sizing only).

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBR	CODE	VAR
AE	A_{exit}	I	Total nozzle exit area		/GENF /(	520)	ACCEL	I	AE
							FH2	I	AE
							IMPUL	I	AE
							PROPB	O	AE
							PROPIN	O	AE
							SOER3	I	AE
FRATED		I	Net rated maximum rocket vacuum thrust	(LBS)	/GENF /(	567)	IMPUL	I	FRATED
							PROPB	O	FRATED
							PROPIN	O	FRATED
FVAC		M	Total vacuum thrust [rocket]	(LBS)	/GENF /(	528)	ACCEL	I	FVAC
							EQUA3	M	FVAC
							FH2	I	FVAC
							IMPUL	M	FVAC
							PROPB	M	FVAC
							PROPIN	M	FVAC
							SOER3	I	FVAC
IFOB		I	Forward or adjoint integration flag = 1 means forward = 2 means adjoint		/XCODES/(	178)	ACCEL	I	IFOB
							BERDCD	I	IFOB
							BLYNE	I	IFOB
							EQUA3	I	IFOB
							IMPUL	I	IFOB
							SPLYNE	I	IFOB
							TOPM	O	IFOB
IRATED		I	Maximum rated ISP	(SEC)	/GENF /(	568)	IMPUL	I	IRATED
ISP	I_{sp}	O	Net vacuum specific impulse		/GENF /(	538)	ACCEL	I	ISP
							IMPUL	O	ISP
ISPF		O	Partial of ISP wrt vacuum thrust		/GENF /(	539)	ACCEL	I	ISPF
							IMPUL	O	ISPF
JPRO		I	Propulsion model option flag		/ARCDAT/(	10)	EQUA3	I	JPRO
							GEINP	I	JPRO
							IMPUL	I	JPRO
							MODELA	I	JPRO
							PROPB	I	JPRO
							PROPIN	I	JPRO
MISP		I	Curve number kISP loss table		/ARCDAT/(	26)	IMPUL	I	MISP
							PROPB	I	MISP
							PROPIN	I	MISP
PA	p_a	I	Atmospheric pressure	(PSF)	/GENF /(	308)	EQUA3	M	DZM
							FH2	I	PA
							IMPUL	I	PA
							OUT	I	PA
							PDBC	I	PA
							SOER3	I	PA
T	T	I	Thrust	(LBS)	/GENF /(	411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SOER3	I	T
XISP	I_{sp}	I	Vacuum specific impulse	(SEC)	/ARCDAT/(	3)	IMPUL	I	XISP
							SIZIN	O	XISP

```
      SUBROUTINE IMPUL                                                     IMPUL
C                                                                          COMN
C      COMPUTES  ISP FOR  ROCKETS                                          COMN
      COMMON/ARCDAT/                                                       ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      ,   ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      ,   ARCDAT
     *XLMAX     ,HDMAX     ,SMOOT     ,ALFMAX    ,PHMAX     ,MAEA      ,   ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      ,   ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      ,   ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        ,   ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                    ARCDAT
     * ,FRATE   , ARCD(9)                                                  RETAP
      DIMENSION  ARCDA(40)                                                 ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                              ARCDAT
      COMMON/GENF/                                                         GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WOC(20)   ,   GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       ,   GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS        ,   GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS        ,   GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     ,   GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     ,   GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         ,   GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       ,   GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   ,   GENF
     *AE        ,FP        ,FPOLD     ,FPO       ,MACHR     ,MACHV     ,   GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                         GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     ,   GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      ,   GENF
     *                      ULFT      ,U_FTV     ,ULFTR     ,ULFTA     ,   GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     ,   GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SIO       ,   GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                       GENF
      COMMON / GENF  /                                                     GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       ,   GENF
     *FRATED    ,IRATED    ,                                               GENF
     *P1        ,P2        ,P3       ,XK1        ,XK2       ,XK3       ,   GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      ,   GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      ,   GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      ,   GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      ,   GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      ,   GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)      GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                  MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                     FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      COMMON /XCODES/                                                      XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JSID(20,2),JPH (20,2),   XCODES
     *JST   (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB     ,   XCODES
     *I2OP      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       ,   XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    ,   XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS        ,   XCODES
     *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE     ,   XCODES
     *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         ,   XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     ,   XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         ,   XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NFHP      ,NPHB      ,   XCODES
     *NCTIN     ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3             XCODES
      FVAC = T + AE*PA                                                     IMPUL
      ISP= XISP                                                            IMPUL
      ISPF = 0.                                                            IMPUL
C                                                                          COMN
C  I   TEST  FOR   DUAL  ENGINE  MODE                                      COMN
      IF(JPRO.EQ.1) GO TO 20                                               AAA      20
C     II   IF  ISP  LOSS  TABLE  NOT INPUT  RETURN                         COMN
      IF(MISP .LE. 0) RETURN                                               IMPUL
      FOR = FVAC/FRATED*100.                                               AAA
      CALL SPLYNE(MISP,FOR,CISP,CISPF)                                     IMPUL
   10 ISP = CISP*IRATED*.01                                                AAA
      IF(IFOB.EQ.1) RETURN                                                 IMPUL
      ISPF= CISPF*.01*IRATED /FRATED                                       AAA
      RETURN                                                               IMPUL
C                                                                          COMN
C     III     EFFECTIVE  ISP   FOR  DUAL ENGINES WITH ONLY ONE             COMN
C                THROTTLED OR  VARYING  WITH  ARC-TIME                     COMN
```

```
   20 FOR= FVAC / FRATED *100.                                          AAA
      CALL ISPRAT(FOR,CISP,CISPF,JUNK,IFOB)                             AAA
      GO TO  10                                                         AAA
      END                                                               IMPUL
```

SUBROUTINE

IPR

Subroutine IPR

Purpose

IPR is a utility routine used for dumping on to the printer either fixed or floating point arrays.

Printed Output

The output of IPR is a BCD name followed by either a 6 column E format array dump or 10 column fixed point array dump.

```
      SUBROUTINE IPR( NAME,DATA,IDATA,N,IND)                           IPR
C                                                                      COMN
C        UTILITY  ARRAY  DUMPER                                        COMN
C        IND =0     FLOATING PT DUMP                                   COMN
C        IND =1     FIXED   PT  DUMP                                   COMN
C        NAME =   HOLLERITH LABEL  FOR DUMP                            COMN
C        DATA =   FLOATING POINT ARRAY OR VARIABLE NAME                COMN
C        IDATA=   FIXED  POINT  ARRAY OR VARIABLE NAME                 COMN
      DIMENSION  DATA(1), IDATA(1)                                     IPR
      PRINT  20, NAME                                                  IPR
      IF(IND.EQ.0) GO  TO  10                                          IPR
      PRINT  30,( IDATA(I),I=1,N)                                      IPR
      RETURN                                                           IPR
   10 PRINT  40,(DATA(I),I=1,N)                                        IPR
      RETURN                                                           IPR
   20 FORMAT (3X,A10)                                                  IPR
   30 FORMAT( 10 I 10 )                                                IPR
   40 FORMAT( 6E17.8)                                                  IPR
      END                                                              IPR
```

FUNCTION
LOMG

FUNCTION LØMG

Purpose

LØMG returns an internally used (steepes + descent) stopping-variable print code given an input code.

Description

LØMG is called from SDINP during the scan of boundary conditions.

```
      FUNCTION  LOMG (K)                                                LOMG
      I=IABS(K)                                                         LOMG
      IF(I.EQ.1) GO TO 10                                               LOMG
      LOMG=I-1                                                          LOMG
      RETURN                                                            LOMG
   10 LOMG=I                                                            LOMG
   20 RETURN                                                            LOMG
      END                                                               LOMG
```

SUBRØUTINE
MAMECØ

Subroutine MAMECØ

Purpose

MAMECØ computes the total aerodynamic moment coefficient and its derivative with respect to Mach number.

Description

MAMECØ is called from Subroutine VT.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALPHA	α	I	Angle of attack	(RAD)	/AECO3 /(	3)	BEROCO	I	ALPHA
							BLGCON	M	ALPHA
							BL2	I	ALPHA
							FNTG	O	ALPHA
							MAMECO	I	ALPHA
							MODELA	M	ALPHA
							MODELB	O	ALPHA
							REU3	O	ALPHA
							VT	I	ALPHA
CM	C_M	O	Moment coefficient		/AECO3 /(	41)	MAMECO	O	CM
							VT	I	CM
CMA	C_{M_α}	I	Moment coefficient slope		/AECO3 /(	42)	EQUA3	M	CMA
							MAMECO	I	CMA
							VT	I	CMA
CMAM	$\partial C_{M_\alpha}/\partial m$	I	See symbol		/AECO3 /(	43)	EQUA3	M	CMAM
							MAMECO	I	CMAM
CMM	$\partial C_M/\partial m$	O	See symbol		/AECO3 /(	44)	MAMECO	O	CMM
							VT	I	CMM
CMO	C_{M_0}	I	Moment coefficient at $\alpha = 0$		/AECO3 /(	45)	EQUA3	I	CMO
							MAMECO	I	CMO
CMOM	$\partial C_{M_0}/\partial m$	I	See symbol		/AECO3 /(	46)	EQUA3	I	CMOM
							MAMECO	I	CMOM

```
      SUBROUTINE MAMECO                                                     MAMECO
      COMMON/GENF/                                                          GENF
     *OMG(20)     ,OMGP(20,2),VARQ(9)   ,TOL(9)     ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)      ,ACON(9)    ,BCON(9)   ,COT1(9,9)  ,DCON(9)   ,DTP       , GENF
     *DTS         ,DT         ,G         ,DPSQ       ,Q         ,QS        , GENF
     *R           ,RE         ,MACH      ,PA         ,RO        ,CS        , GENF
     *VNU         ,PAR        ,ROR       ,CSR        ,VNR       ,SUMSQ     , GENF
     *SVSQ        ,TIMEPH     ,TIMES     ,TOP        ,TDS       ,TR(9)     , GENF
     *TST(20)     ,TPH   (20),DIS(20)   ,DIP(20)    ,T         ,W         , GENF
     *TLP1(20)    ,TLS1  (20),DIP1(20)  ,DIS1(20)   ,TIME      ,DMP       , GENF
     *TIMPR       ,LIFT       ,DRAG      ,TAX        ,TBURN     ,TBU(20)   , GENF
     *AE          ,FP         ,FPOLD     ,FPD        ,MACHR     ,MACHV     , GENF
     *QR          ,QV         ,FVAC      ,LIFTV      ,                       GENF
     *LIFTR       ,LIFTA      ,           DRAGV      ,DRAGR     ,DRAGA     , GENF
     *             LIFTM      ,DBR       ,DB         ,ISP       ,ISPF      , GENF
     *                         ULFT      ,ULFTV      ,ULFTR     ,ULFTA     , GENF
     *XMCG        ,XMCGV      ,XMCGR     ,XMCGA      ,XMCGM     ,CODAE     , GENF
     *CULFT       ,CT         ,CALPHA    ,CDE        ,DELTAE    ,SID       , GENF
     *COD         ,SIDAE      ,XCG       ,ZCG        ,XJ                     GENF
      COMMON / GENF /                                                       GENF
     *XJV         ,XJR        ,GH        ,GAMMAD     ,XKG       ,XKP       , GENF
     *FRATED      ,IRATED     ,                                             GENF
     *P1          ,P2         ,P3       ,XK1         ,XK2       ,XK3       , GENF
     *XK1T        ,XK2T       ,XK3T      ,XK1D       ,XK2D      ,XK3D      , GENF
     *XK1A        ,XK2A       ,XK3A      ,XK1V       ,XK2V      ,XK3V      , GENF
     *XK1G        ,XK2G       ,XK3G      ,XK1P       ,XK2P      ,XK3P      , GENF
     *XK1R        ,XK2R       ,XK3R      ,XK1O       ,XK2O      ,XK3O      , GENF
     *XK1U        ,XK2U       ,XK3U      ,XK1M       ,XK2M      ,XK3M      , GENF
     *PV          ,PG         ,PP        ,PR         ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                      FRAT
      DIMENSION TPH1(10),TST1(10)                                           GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                    GENF
      COMMON/AECO3/                                                         AECO3
     *APHO        ,APHR       ,ALPHA     ,VDA        ,GDA       ,PDA       , AECO3
     *SINA        ,COSA       ,PHIO      ,PHID       ,PHI       ,SINPHI    , AECO3
     *COSPHI      ,GDPH       ,PDPH      ,XLAMA(9)   ,XLAMP(9)  ,CDO       , AECO3
     *CDOM        ,CLO        ,FK        ,XCGM       ,ZCGM      ,CLOM      , AECO3
     *CM          ,CMA        ,CMAM ,CMM ,CMO        ,CMOM      ,FKM       , AECO3
     *CLAM        ,CL         ,CLA       ,CLM        ,                       AECO3
     *CD          ,CDA        ,CDM                                          AECO3
      CM = CMO + ALPHA*CMA                                                  MAMECO
      CMM = CMOM + ALPHA*CMAM                                               MAMECO
      RETURN                                                                MAMECO
      END                                                                   MAMECO
```

SUBRØUTINE
MØDELA

Subroutine MODELA

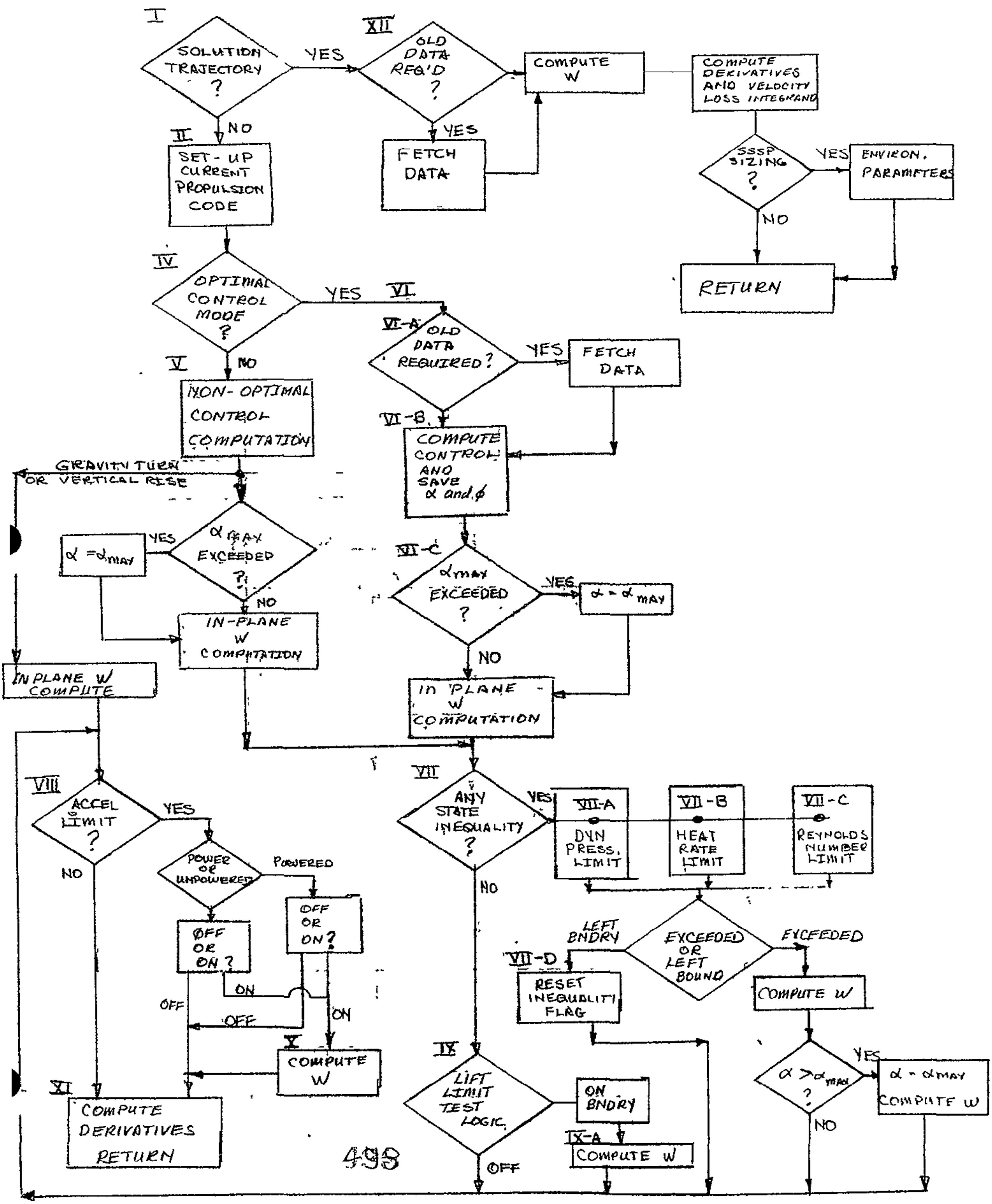
Purpose

MODELA governs the computation of the control vector and sets up the derivative calculation during the forward trajectory integration.

Description

The way the control vector is calculated depends on options in the program and also on control and state inequality constraints. MODELA does not calculate the control vector (BLGCØN does), but it does test the result of either optimizing or non-optimal control computations to determine if instantaneous constraints have been violated. If constraints are violated, the control mode is instantaneously changed and control is recalculated on the appropriate constraint boundary. After the bounded control is computed, the derivatives are calculated and returned to the calling routine, FNTG.

SUBROUTINE MODELA



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALFMAX	α_{MAX}	I	Maximum angle of attack	(DEG)	/ARCDAT/(	16)	INBVAD	M	ALFMAX
							MODELA	I	ALFMAX
							MODELB	I	ALFMAX
ALFSAV		W	Angle of attack prior to control boundry		/MODELA/(*	)	MODELA	W	ALFSAV
ALPHA	α	M	Angle of attack	(RAD)	/AEC03 /(	3)	BEROCO	I	ALPHA
							BLGCON	M	ALPHA
							BL2	I	ALPHA
							FNTG	O	ALPHA
							MAMECO	I	ALPHA
							MODELA	M	ALPHA
							MODELB	O	ALPHA
							REU3	O	ALPHA
							VT	I	ALPHA
APHR	α	M	Angle of attack	(DEG)	/AEC03 /(	2)	AGETB3	O	APHR
							AST3	M	APHR
							BEROCO	I	APHR
							BLGCON	O	APHR
							GUI3A	M	APHR
							MODELA	M	APHR
							MODELB	I	APHR
							MTX3A	O	APHR
							OUT	I	APHR
CALPHA		M	Constant value of angle-of-attack	(RAD)	/GENF /(	552)	BL2	I	CALPHA
							MODELA	M	CALPHA
							MODELB	M	CALPHA
COSGAM	$\cos(\gamma)$	I	See symbol		/STATE3/(	687)	ACCEL	I	COSGAM
							BL4	I	COSGAM
							BL8	I	COSGAM
							DER3A	I	COSGAM
							EQUA3	O	COSGAM
							MODELA	I	COSGAM
							MODELB	I	COSGAM
							OUT	I	COSGAM
							PDBC	I	COSGAM
							PDY3A	I	COSGAM
COSPHI	$\cos\phi$	M	See symbol		/AEC03 /(	13)	ACCEL	I	COSPHI
							BL4	I	COSPHI
							GUI3A	M	COSPHI
							MODELA	M	COSPHI
							MODELB	M	COSPHI
							OUT	I	COSPHI
COSPSI	$\cos(\psi)$	I	See symbol		/STATE3/(	705)	BL4	I	COSPSI
							BL7	I	COSPSI
							BL8	I	COSPSI
							DER3A	I	COSPSI
							EQUA3	O	COSPSI
							MODELA	I	COSPSI
							MODELB	I	COSPSI
							PDBC	I	COSPSI
							PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol		/STATE3/(	707)	BL4	I	COSRHO
							BL7	I	COSRHO
							BL8	I	COSRHO
							DER3A	I	COSRHO
							EQUA3	M	COSRHO
							MODELA	I	COSRHO
							MODELB	I	COSRHO
							OUT	I	COSRHO
							PDBC	I	COSRHO
							PDY3A	I	COSRHO
COS2GM	$\cos(2\gamma)$	O	See symbol		/STATE3/(	757)	MODELA	O	COS2GM
							MODELB	O	COS2GM
COS2RO	$\cos(2\rho)$	O	See symbol		/STATE3/(	756)	BL4	I	COS2RO
							BL7	M	COS2RO
							BL8	M	COS2RO
							MODELA	O	COS2RO
							MODELB	O	COS2RO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
CULFT		O	Constant value of ULFT	(LBS)	/GENF /(	550)	BL3	I	CULFT
							MODELA	O	CULFT
G	g	I	Gravitational attraction	(FT/SEC**2)	/GENF /(	301)	BL4	I	G
							BL7	I	G
							BL8	I	G
							DER3A	I	G
							EQUA3	M	G
							MODELA	I	G
							MODELB	I	G
							POY3A	I	G
							SDER3	I	G
							SDINP	M	G
GMAX	G_{MAX}	I	Maximum total acceleration g load		/ARCDAT/(	12)	BL5	I	GMAX
							FH3	I	GMAX
							MODELA	I	GMAX
							PROPB	I	GMAX
							PROPIN	I	GMAX
GMDOT	$\dot{\gamma}*$	I	Pitch rate	(DEG/SEC)	/ARCDAT/(	15)	DER3A	I	GMDOT
							MODELA	I	GMDOT
							MODELB	I	GMDOT
							PROPB	I	GMDOT
							PROPIN	I	GMDOT
IARC		I	Arc number		/XCODES/(	146)	ADICB3	I	IARC
							ADIC3A	I	IARC
							ADJUST	I	IARC
							AST3	I	IARC
							BNTG	M	IARC
							ENVPRM	I	IARC
							FNTG	M	IARC
							GETIT	I	IARC
							MODELA	I	IARC
							PROPB	I	IARC
							PROPIN	I	IARC
							REU3	I	IARC
							SDINP	M	IARC
							STAU	I	IARC
							STP3	I	IARC
							TRTOSZ	I	IARC
ISTART		O	Initialization and divergence flag		/XCODES/(	147)	AST3	O	ISTART
							BLGCON	O	ISTART
							BLYNE	O	ISTART
							FNTG	I	ISTART
							MODELA	O	ISTART
							PROPIN	O	ISTART
							REU3	I	ISTART
							TEST	M	ISTART
							TOPM	M	ISTART
ITER		I	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION		/XCODES/(	149)	AST3	I	ITER
							FNTG	I	ITER
							GETIT	I	ITER
							MODELA	I	ITER
							OUT	I	ITER
							PAYO2	M	ITER
							PROPIN	I	ITER
							TEST	M	ITER
							TOPM	M	ITER
JGII		I	Control option		/XCODES/(	195)	ACCEL	I	JGII
							BNTG	O	JGII
							DER3A	I	JGII
							FNTG	M	JGII
							GUI3A	I	JGII
							MODELA	I	JGII
							MODELB	I	JGII
							MTX3A	I	JGII
							POY3A	I	JGII

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
JK		I	Integration routine flag tells which derivative evaluation in Runge-Kutta cycle	/XCODES/(	151)	ADIC3A	M	JK
						BNTG	I	JK
						MODELA	I	JK
						PAYO2	M	JK
						RKTA3A	M	J
						RKTB3A	M	J
JPRO		I	Propulsion model option flag	/ARCDAT/(	10)	EQUA3	I	JPRO
						GEINP	I	JPRO
						IMPUL	I	JPRO
						MODELA	I	JPRO
						PROPB	I	JPRO
						PROPIN	I	JPRO
JPRP		I	Propulsion flag for different rocket options	/XCODES/(	194)	ACCEL	I	JPRP
						DER3A	I	JPRP
						EQUA3	I	JPRP
						MODELA	I	JPRP
						PDY3A	I	JPRP
						PROPB	O	JPRP
						PROPIN	O	JPRP
JP1		M	Option flag for first governing equation	/XCODES/(	217)	AGETB3	M	JP1
						AST3	M	JP1
						MODELA	M	JP1
						MODELB	I	JP1
						PROPB	O	JP1
						PROPIN	O	JP1
JP2		I	Option flag for second governing equation	/XCODES/(	218)	MODELA	I	JP2
						MODELB	I	JP2
						PROPB	O	JP2
						PROPIN	O	JP2
JP3		M	Option flag for third governing equation	/XCODES/(	219)	AGETB3	O	JP3
						AST3	M	JP3
						MODELA	M	JP3
						MODELB	I	JP3
						OUT	I	JP3
						PROPIN	O	JP3
JTYP		I	Sizing. Flag.	/SIZING/(	313)	FNTG	I	JTYP
						GEINP	O	JTYP
						MODELA	I	JTYP
						PADS1	I	JTYP
						PROPIN	I	JTYP
						SIZIN	I	JTYP
						TRTOSZ	I	JTYP
K		I	Storage retrieval flag indicates end of arc, phase, or data.	/XCODES/(	156)	AST3	O	K
						FNTG	O	K
						GETIT	M	K
						MODELA	I	K
						SDINP	M	K
OCORHO	$\omega\times$COSRHO	I	See symbol	/STATE3/(	708)	DER3A	I	OCORHO
						EQUA3	M	OCORHO
						MODELA	I	OCORHO
						MODELB	I	OCORHO
						PDBC	I	OCORHO
						PDY3A	I	OCORHO
OCORO2	$\omega\times$OCORHO	I	See symbol	/STATE3/(	709)	DER3A	I	OCORO2
						EQUA3	O	OCORO2
						MODELA	I	OCORO2
						MODELB	I	OCORO2
						PDY3A	I	OCORO2

8 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
OMGZ	ω	I	Earth rotation rate	(RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
							CRASH	I	OMEGA
							DER3A	I	OMGZ
							EQUA3	I	OMGZ
							GEINP	I	OMGZ
							MODELA	I	OMGZ
							MODELB	I	OMGZ
							PDBC	I	OMGZ
							PDY3A	I	OMGZ
							SDINP	I	OMGZ
							TOPM	I	OMGZ
PHI	ϕ	M	Bank angle	(RAD)	/AECO3 /(	11)	GUI3A	M	PHIR
							MODELA	M	PHI
							MODELB	M	PHI
							OUT	I	PHI
PHID	ϕ	M	BanK angle	(DEG)	/AECO3 /(	10)	AGETB3	O	PHID
							AST3	M	PHID
							GUI3A	M	PHID
							MODELA	M	PHID
							MODELB	I	PHID
							MTX3A	O	PHID
							OUT	I	PHID
PHISAV		W	Bank angle prior to control boundry		/MODELA/(*	)	MODELA	W	PHISAV
R	R	I	Radial distance from earth center to vehicle	(FT)	/GENF /(	305)	BL4	I	R
							BL7	I	R
							BL8	I	R
							DER3A	I	R
							EQUA3	M	R
							MODELA	I	R
							MODELB	I	R
							PDBC	I	R
							PDY3A	I	R
							TRTOSZ	I	R
RAD		I	Radian to angle conversion, 57 29577951		/DATA /(	2)	BEROCO	I	DEG
							BLGCON	I	RAD
							ENVPRM	I	RAD
							EQUA3	I	RAD
							FNTG	I	RAD
							GUI3A	I	RAD
							MODELA	I	RAD
							MTX3A	I	RAD
							OUT	I	RAD
							PADS1	D	RAD
							SDINP	I	RAD
							TRTOSZ	I	RAD
RDI		I	Angle to radian conversion, 01745329252		/DATA /(	3)	BLICO	I	RDI
							DER3A	I	RDI
							FNTG	I	RDI
							GUI3A	I	RDI
							MODELA	I	RDI
							MODELB	I	RDI
							PADS1	D	RDI
							PROPB	I	RDI
							PROPIN	I	RDI
							REU3	I	RDI
							SDINP	I	RDI
							SOMG	I	RDI
SINGAM	$\sin(\gamma)$	I	See symbol		/STATE3/(	688)	BL4	I	SINGAM
							BL7	I	SINGAM
							BL8	I	SINGAM
							DER3A	I	SINGAM
							EQUA3	O	SINGAM
							MODELA	I	SINGAM
							MODELB	I	SINGAM
							PDBC	I	SINGAM
							PDY3A	I	SINGAM
							SDER3	I	SINGAM

6 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
SINPHI	$\sin\phi$	M	See symbol	/AEC03 /(	12)	ACCEL	I	SINPHI
						BL9	I	SINPHI
						MODELA	M	SINPHI
						MODELB	M	SINPHI
						OUT	I	SINPHI
SINPSI	$\sin(\psi)$	I	See symbol	/STATE3/(	704)	BL4	I	SINPSI
						BL7	I	SINPSI
						BL8	I	SINPSI
						DER3A	I	SINPSI
						EQUA3	O	SINPSI
						MODELA	I	SINPSI
						MODELB	I	SINPSI
						PDBC	I	SINPSI
						PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol	/STATE3/(	706)	BL4	I	SINRHO
						BL7	I	SINRHO
						BL8	I	SINRHO
						DER3A	I	SINRHO
						EQUA3	O	SINRHO
						MODELA	I	SINRHO
						MODELB	I	SINRHO
						OUT	I	SINRHO
						PDBC	I	SINRHO
						PDY3A	I	SINRHO
SIN2RO	$\sin(2\rho)$	O	See symbol	/STATE3/(	755)	BL4	I	SIN2RO
						BL7	M	SIN2RO
						BL8	M	SIN2RO
						MODELA	O	SIN2RO
						MODELB	O	SIN2RO
SQRT		F	Square root function	/SQRT /($	)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTOE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAYOZ	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
TBURN	t_b	I	Rocket burn initiation time on forward trajectory[sd]	/GENF /(	499)	EQUA3	I	TBURN
						MODELA	I	TBURN
						PROPB	O	TBURN
						PROPIN	M	TBURN
TIME	t	I	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
TPRP		W	Burn time	/MODELA/(*	)	MODELA	W	TPRP

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ULFT	L_U	I	Untrimmed aero.lift		/GENF /(	540)	BL3	I	ULFT
							MODELA	I	ULFT
							VT	M	ULFT
V	v	I	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	Y
							STP3	I	VAR
							TOPM	D	KWOW
							VREF3	M	V
VAR	v	I	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	Y
							STP3	I	VAR
							TOPM	D	KWOW
							VREF3	M	V
XKGAM	k_γ	M	Algebraic equation used in vertical rise and pitchover		/GENF /(	565)	BL4	I	XKG
							MODELA	M	XKGAM
							MODELB	M	XKGAM
XKPS	k_ψ	M	Algebraic equation used in vertical rise and pitchover		/GENF /(	566)	BL4	I	XKP
							MODELA	M	XKPS
							MODELA	I	XKPSI
							MODELB	M	XKPS
							MODELB	I	XKPSI
XKPSI	k_ψ	I	Algebraic equation used in vertical rise and pitchover		/GENF /(	566)	BL4	I	XKP
							MODELA	M	XKPS
							MODELA	I	XKPSI
							MODELB	M	XKPS
							MODELB	I	XKPSI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
XK1		I	First control vector governing equation value corresponds to error in thrust	/GENF /(	572)	BLGCON	I	XK1
						FH1	O	XK1
						FH2	O	XK1
						FH3	O	XK1
						FH4	O	XK1
						MODELA	I	XK1
XK1T		I	Partial of governing equation wrt state or control vector component	/GENF /(	575)	BLGCON	I	XK1T
						FH1	O	XK1T
						FH2	O	XK1T
						FH3	M	XK1T
						FH4	O	XK1T
						MODELA	I	XK1T
XK3		I	Third control vector governing equation value Corresponds to error in algebraic equation involving α.	/GENF /(	574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3
XLMAX	L_{MAX}	I	Maximum aerodynamic lift (LBS)	/ARCDAT/(	13)	MODELA	I	XLMAX
.UN06.		O	File of all output data	/.UN06 /($	)	BLICO	O	.UN06.
						BNDRYC	O	.UN06.
						CRASH	O	.UN06.
						FRENCH	O	UN06.
						FXDAT	O	.UN06.
						GEINP	O	UN06.
						HUNT	O	.UN06.
						INEDIT	O	.UN06.
						ITER8	O	.UN06.
						MODELA	O	.UN06
						MOMJ	O	.UN06.
						MPSI	O	.UN06.
						OUT	O	.UN06.
						PAYD2	O	UN06.
						PRINT	O	.UN06.
						PRINTV	O	.UN06.
						PRINTW	O	.UN06
						PRITEQ	O	UN06
						PRITVA	O	UN06.
						PROPIN	O	.UN06.
						PROTHR	O	UN06
						PRWTSM	O	UN06.
						RANGE	O	.UN06.
						S	O	UN06.
						SOINP	O	UN06.
						SIZE	O	UN06
						SIZIN	O	UN06.
						SIZOUT	O	UN06.
						SOLVE	O	UN06
						SPLICO	O	UN06.
						SPLIZ	O	.UN06.
						SPLYNE	O	UN06.
						SSSP	O	.UN06
						STAU	O	UN06.
						STPIT	O	UN06.
						SUMOUT	O	.UN06
						TABIN	O	UN06.
						TEST	O	UN06
						VEHDF	O	.UN06
						WTSCH	O	UN06.
						WTVOL	O	.UN06

```
      SUBROUTINE  MODELA (ITT,STT)                                      MODELA
C        THIS ROUTINE  CONTROLS COMPUTATION OF DECISION  VECTOR         MODELA
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)      ,SVY(10) , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM     ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2    , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2     ,                                  STATE3D
     *VDV        ,GDV        ,RDV        ,MDV        ,PDV        ,ODV       , STATE3D
     *UDV        ,VDG        ,GDG        ,RDG        ,PDG        ,ODG       , STATE3D
     *UDG        ,VDR        ,GDR        ,MDR        ,PDR        ,ODR       , STATE3D
     *UDR        ,VDM        ,GDM        ,MDM        ,PDM        ,VDP       , STATE3D
     *GDP        ,PDP        ,ODP        ,UDP        ,VDO        ,GDO       , STATE3D
     *PDO        ,UDO        ,HTDV       ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                     STATE3D
      COMMON /XCODES/                                                   XCODES
     *ITO    (9),ICOR  (20),ITI        ,INTB       ,JGIO(20,2),JPH (20,2), XCODES
     *JST   (20)             ,NCNST      ,NSB        ,NSAB       ,NICNB     , XCODES
     *I2OP       ,ICOP       ,IFAW       ,IFAR       ,IFB        ,IND       , XCODES
     *IOPEN      ,IPH        ,ISPH       ,ISST       ,IARC       ,ISTART    , XCODES
     *ITCT       ,ITER       ,IVAR       ,JK         ,JPS        ,JS        , XCODES
     *KOP        ,KPST       ,K          ,KST        ,NAD        ,NCASE     , XCODES
     *NCN        ,NEQB       ,NEQ        ,NOP        ,NPH        ,N         , XCODES
     *NST        ,IPST       ,IPRINT     ,ISTN       ,IPHN       ,ISTNB     , XCODES
     *IPHNB      ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP      ,L         , XCODES
     *IFOB       ,NB         ,LB         ,MB         ,NPHP       ,NPHB      , XCODES
     *NCTIN      ,NEQF  ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,  EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2),  EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),  EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)             EQUV3
      REAL M,MU,MD                                                      EQUV3
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)    ,TOL(9)     ,SVAR(10)   ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)    ,DTP       , GENF
     *DTS        ,DT         ,G          ,DPSQ       ,Q          ,QS        , GENF
     *R          ,RE         ,MACH       ,PA         ,RO         ,CS        , GENF
     *VNU        ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)     , GENF
     *TST(20)    ,TPH    (20),DIS(20)    ,DIP(20)    ,T          ,W         , GENF
     *TLP1(20)   ,TLS1   (20),DIP1(20)   ,DIS1(20)   ,TIME       ,OMP       , GENF
     *TIMPR      ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)   , GENF
     *AE         ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV     , GENF
     *QR         ,QV         ,FVAC       ,LIFTV      ,                        GENF
     *LIFTR      ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA     , GENF
     *            LIFTM      ,DBR        ,DB         ,ISP        ,ISPF      , GENF
     *                        ULFT       ,ULFTV      ,ULFTR      ,ULFTA     , GENF
     *XMCG       ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE     , GENF
     *CULFT      ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SID       , GENF
     *COD        ,SIDAE      ,XCG        ,ZCG        ,XJ                      GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP       , GENF
     *FRATED     ,IRATED     ,                                            GENF
     *P1         ,P2         ,P3        ,XK1         ,XK2        ,XK3       , GENF
     *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D      , GENF
     *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V      , GENF
     *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P      , GENF
     *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O      , GENF
     *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M      , GENF
     *PV         ,PG         ,PP         ,PR         ,PO         ,OPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF       ,EJ         ,XISP       ,TMULT      ,DTNC       ,DTPI      , ARCDAT
     *IATM       ,IMODE      ,JAER       ,JPRO       ,QMAX       ,GMAX      , ARCDAT
     *XLMAX      ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX      ,MAEA      , ARCDAT
     *MAEB       ,MAEC       ,MAED       ,MAEE       ,MAEF       ,MAEG      , ARCDAT
     *MT         ,MISP       ,MXCG       ,MZCG       ,MWDA       ,MWDB      , ARCDAT
     *MDB        ,XCGR       ,ZCGR       ,XE         ,ZE         ,XT        , ARCDAT
     *DREF       ,MCND       ,RHOB       ,QMULT      ,REMAX                   ARCDAT
     * ,FRATE    , ARCD(9)                                              RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
```

```
      EQUIVALENCE(SREF,ARCDA)                                            ARCDAT
      COMMON/AECO3/                                                      AECO3
     *APHD      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       , AECO3
     *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI    , AECO3
     *COSFHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       , AECO3
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLGM      , AECO3
     *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM       , AECO3
     *CLAM      ,CL        ,CLA       ,CLM       ,                       AECO3
     *CD        ,CDA       ,CDM                                          AECO3
      COMMON/GLOBAL/                                                     GLOBAL
     *GR        ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM         GLOBAL
     *,JJOP(10) ,IFATAL    ,NARC      ,NBRAN     ,NFARC     ,ID(4)       GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                                GLOBAL
     *,GM       ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)       GLOBAL
     *,ITPSO ,KSOL     ,KGLOBL(8)                                        RETAP
      REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                            SIZING
      COMMON /SIZING/                                                    SIZING
C            PHASE II SIZING PARAMERERS                                  SIZING
     *TZ,      VV(3),     QP(14),    EROR,     PZ(5),     VQ,      SW(20), SIZING
     *SV(28),  SQ(37,5),  SE(11),    TLAT,     TLNG,                     SIZING
C           PHASE I SIZING PARAMERERS                                    SIZING
     *WBO,     WLOO,      DWEB,      DWEO,     TOLWT,     WPB,     TWRAT2, SIZING
     *BK1,     BK2,       BK3,       BK4,      ISIZE,     TRAFLG,  TWRATO, SIZING
     *OK1,     OK2,       OK3,       OK4,      PRFLG,     IPASS,   IPSMAX, SIZING
     *AEXIT,   TVACO,     NO,        WFO,      IDVEL,     ISPO,    ISPB,   SIZING
     *XPL,     TVACB,     NNB,       WEO,      WEB,       WO,      WLO,    SIZING
     *DVO,     DVB,       MUB,       MUO,      VSTG,      WPO              SIZING
     *,JTYP,   BECO  ,    BSTG ,     ORBI,     ITNBW , ITNOW  ,            SIZING
     * SVDPSQ  , SVOCON  ,IHUNT   ,IOPSTG  ,ISZD(19)                     UH
      COMMON/DATA/                                                       DATA
     *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      , DATA
     *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4        DATA
      EQUIVALENCE(JIN,ILAB(1))                                           P014
      EQUIVALENCE (INEQFL(20),INQF),(XKPS,XKP),(XKGAM,XKG)               MODELA
      EQUIVALENCE (XKPSI,XKP)                                            IO
C     I     SOLUTION TRAJECTORY  TEST                                    MODELA
      IF(ITER.EQ.3)  GO TO 340                                           MODELA
C      II  SET  UP  PROPULSION CODE                                      MODELA
      IF(JPRP-2) 30,10,20                                                MODELA
   10 TPRP = TIME - TBURN                                                MODELA
      JP1 = 2                                                            MODELA
      GO TO 40                                                           MODELA
   20 TPRP = TIME -TBURN                                                 MODELA
      IF(TPRP GT 0) GO  TO  40                                           MODELA
      JIN =0                                                             MODELA
      JP1 = 2                                                            MODELA
      GO TO  40                                                          MODELA
   30 IF(JPRO NE 2) JP1=1                                                COMN
   40 CONTINUE                                                           MODELA
C     IV      TEST   FOR       OPTIMAL  CONTROL                          MODELA
   50 IF(JGII GT 10)    GO  TO  130                                      MODELA
C       V     COMPUTE  NON-OPTIMAL   CONTROL                             MODELA
      GO TO(60,60,60,999,999,70,80,100),JGII                             MODELA
   60 CALL  GUID                                                         MODELA
      ALPHA=APHR*RDI                                                     P014
       CALPHA=ALPHA                                                      P014
      PHI=PHID*RDI                                                       P014
      SINPHI=SIN(PHI)                                                    P014
      COSPHI=COS(PHI)                                                    P014
      GO  TO  110                                                        MODELA
   70 APHR = 0                                                           MODELA
      ALPHA= 0                                                           MODELA
      PHI= 0.                                                            MODELA
      CALPHA=0                                                           P014
      PHID=0.                                                            P014
      SINPHI=0.                                                          P014
      COSPHI=1.                                                          P014
      GO TO 110                                                          MODELA
C       V-A  GRAVITY  TURN                                               MODELA
   80 JP3 = 6                                                            MODELA
   90 CALL   BLGCON(JP1,JP2,JP3)                                         MODELA
```

```
          PHID=PHI*RAD                                                    P014
          GO  TO  240                                                     MODELA
C       V-B VERTICAL  RISE  OR  PITCHOVER                                 MODELA
      100 JP3=4                                                           MODELA
          XKPS =SINRHO*SINPSI*(R*OCORO2+V*V*COSGAM*COSGAM/(R*COSRHO))   + MODELA
        1 2 *V*(OMGZ*SINRHO*COSGAM-OCORHO* COSPSI* SINGAM)                MODELA
          XKGAM = OCORHO*(2.*V*SINPSI+ R*OMGZ *(COSRHO*COSGAM+ SINRHO*COSPSI MODELA
        1*SINGAM)) + COSGAM*(V*V/R- G)- V*RDI*GMDOT                       MODELA
          D= SQRT(XKGAM*XKGAM + XKPS*XKPS)                                MODELA
          SINPHI = SIGN(1 ,XKGAM)* XKPSI/D                                JULY28
          COSPHI= ABS(XKGAM)/ D                                           MODELA
          PHI= ATAN2(SINPHI,COSPHI)                                       MODELA
          SIN2RO= 2 *SINRHO* COSRHO                                       MODELA
          COS2RO=(COSRHO +SINRHO)*(COSRHO-SINRHO)                         MODELA
          COS2GM =(COSGAM -SINGAM)*(COSGAM + SINGAM)                      MODELA
          GO TO 90                                                        MODELA   90
C       V-C   TEST  FOR  ALFMAX  ON A  NON-OPTIMAL CONTROL MODE           MODELA
      110 JP3= 2                                                          MODELA
          IF(ALFMAX.EQ 0) GO TO 120                                       MODELA   120
          IF(ABS(APHR) GT ALFMAX)APHR = SIGN(ALFMAX,APHR)                 MODELA
          CALPHA = APHR*RDI                                               MODELA
          ALPHA=CALPHA                                                    P014
      120 CALL BLGCON(JP1,JP2,JP3)                                        MODELA
          GO TO  170                                                      MODELA   170
C   VI      ON  AN  OPTIMAL CONTROL  MODE                                 MODELA
C  VI-A     TEST  WHETHER  OLD DATA  IS  REQUIRED                         MODELA
      130 IF(TIME EQ.STT AND.K EQ 1 AND.ITT.EQ.IARC) GO TO 140            MODELA   140
          CALL  GETIT                                                     MODELA
C  VI-B     COMPUTE   CONTROL                                             MODELA
          CALL  MTX1                                                      MODELA
          ALFSAV=APHR                                                     APR27
          PHISAV=PHID                                                     APR27
          GO TO 142                                                       APR27    142
C  VIC       TEST  FOR   ALFMAX                                           MODELA
      140 APHR=ALFSAV                                                     APR27
          PHID=PHISAV                                                     APR27
      142 IF(ALFMAX EQ.0.) GO TO 150                                      APR27    150
          IF(ABS(APHR) GT ALFMAX)  APHR = SIGN(ALFMAX,APHR)               MODELA
      150 CALPHA= APHR*RDI                                                MODELA
          ALPHA=CALPHA                                                    P014
          PHI=PHID*RDI                                                    MODELA
          SINPHI= SIN(PHI)                                                MODELA
          COSPHI= COS(PHI)                                                MODELA
          JP3 =2                                                          MODELA
      160 CALL  BLGCON(JP1,JP2,JP3)                                       MODELA
C       VII      TEST  FOR    STATE  INEQUALITIES                         MODELA
      170 IF(INQF EQ 0)  GO TO  290                                       MODELA   290
          IF(INQF-8)  180,190,200                                         MODELA   180 190 200
C     VII-A  TEST  FOR  BNDRY  ON  DVN  PRES.                             MODELA
      180 CALL  BL7000                                                    MODELA
          GO TO  210                                                      MODELA   210
C   VII-B   TEST  FOR  BNDRY  ON  HEAT RATE                               MODELA
      190 CALL  BL8000                                                    MODELA
          GO TO  210                                                      MODELA   210
C  VII-C      TEST  FOR  BNDRY  ON  REY  NO                               MODELA
      200 CALL  BL9000                                                    MODELA
      210 IF(XK3) 220,230,230                                             MODELA   220 230
C  VII-D       HAS  LEFT  BNDRY                                           MODELA
      220 INQF =0                                                         MODELA
          JP3= 2                                                          MODELA
          GO TO 240                                                       MODELA   240
C  VII-E     STILL  ON  BNDRY.                                            MODELA
      230 JP3 = INQF                                                      MODELA
          IF(ALPHA*COSPHI.GE.0.) GO TO 232                                APR27    232
          ALPHA=SIGN( 2,COSPHI)                                           APR27
      232 CONTINUE                                                        APR27
          CALL BLGCON(JP1,JP2,JP3)                                        MODELA
C  VII-F     TEST  FOR   ALFMAX                                           MODELA
```

20 OCT 72 G 01-46

```
      IF(ALFMAX EQ 0 ) GO TO 240                                        MODELA
      APHR=  ALPHA*RAD                                                  MODELA
      IF(ABS(APHR) LE ALFMAX)  GO  TO 240                               MODELA
      APHR  = SIGN(ALFMAX,APHR)                                         MODELA
      CALPHA =  APHR*RDI                                                MODELA
      ALPHA=CALPHA                                                      P014
      JP3 =2                                                            MODELA
      CALL BLGCON(JP1,JP2,JP3)                                          MODELA
C  VIII    TEST  FOR  ACCEL  LIMIT                                      MODELA
  240 IF(GMAX EQ.0.)  GO  TO  320                                       MODELA
      IF(JPRP - 2)   250,270,270                                        MODELA
C VIII-A   UNPOWERED   ACCELERATION LIMIT  TEST                         MODELA
  250 CALL BL5000                                                       MODELA
      IF(XK3)   320,320,260                                             MODELA
  260 JP3 =5                                                            MODELA
      GO TO  310                                                        MODELA
C  VIII-B   POWERED  ACCELERATION  LIMIT                                MODELA
  270 IF(JIN.GT 0) GO TO 320                                            MODELA
      CALL  FH3000                                                      MODELA
      IF(XK1) 320,320,280                                               MODELA
  280 CALL FH3001                                                       APR27
      IF(XK1T.LT 0.) GO TO 320                                          APR27
      JIN=1                                                             APR27
      WRITE(6,285)  TIME                                                P014
  285 FORMAT(2X,42H***HIT POWERED ACCELERATION LIMIT AT TIME= F10 4)    P014
      JP1 = 3                                                           MODELA
      GO TO  310                                                        MODELA
C     IX        TEST  OF  LIFT  LIMIT                                   MODELA
  290 IF(XLMAX EQ 0 ) GO TO 240                                         MODELA
      CULFT=SIGN(XLMAX,ULFT)                                            P014
      CALL  BL3000                                                      MODELA
      IF(XK3) 240,240,300                                               MODELA
  300 JP3 = 3                                                           MODELA
C     IX-A      COMPUTE  CONTROL  FOR  LIFT  LIMIT                      MODELA
      CALL BLGCON(JP1,JP2,JP3)                                          MODELA
      GO TO  240                                                        MODELA
C  X   FINAL  CLEANUP                                                   MODELA
  310 CALL  BLGCON(JP1,JP2,JP3)                                         MODELA
C  XI     DERIVATIVE   COMPUTATIONS                                     MODELA
  320 CONTINUE                                                          P014
      APHR=ALPHA*RAD                                                    P014
      IF(ABS(APHR) GT 240 .OR.ABS(VAR(2)).GT 7.) ISTART =6              P014
      IF(VAR(1) LT 0 ) ISTART=6                                         P014
      IF(ABS(VAR(5)) GE.7.) ISTART=6                                    P014
      IF(JK.NE.2) GO TO 330                                             P014
      CALL  CORVAR                                                      MODELA
      RETURN                                                            MODELA
  330 CALL  DER                                                         MODELA
      RETURN                                                            MODELA
C     XII   SOLUTION  TRAJECTORY  LOGIC                                 MODELA
  340 IF(TIME.EQ STT AND.K EQ.1.AND.ITT EQ.IARC) GO TO 360              MODELA
      CALL  GETIT                                                       MODELA
      IF(JP3.NE 4) GO TO 350                                            MODELA
      XKPS =SINRHO*SINPSI*(R*DCORD2+V*V*COSGAM*COSGAM/(R*COSRHO))     + MODELA
     1 2 *V*(OMGZ*SINRHO*COSGAM-DCORHO* COSPSI* SINGAM)                 MODELA
      XKGAM = DCORHO*(2 *V*SINPSI+ R*OMGZ *(COSRHO*COSGAM+ SINRHO*COSPSI MODELA
     1*SINGAM)) + COSGAM*(V*V/R- G)- V*RDI*GMDOT                        MODELA
      D= SQRT(XKGAM*XKGAM + XKPS*XKPS)                                  MODELA
      SINPHI = SIGN(1 ,XKGAM)* XKPSI/D                                  JULY28
      COSPHI= ABS(XKGAM)/ D                                             MODELA
      PHI= ATAN2( SINPHI,COSPHI)                                        MODELA
      PHID = PHI *RAD                                                   MODELA
      COS2RO=(COSRHO - SINRHO)*(COSRHO + SINRHO)                        MODELA
      SIN2RO=  2 *SINRHO*COSRHO                                         MODELA
      COS2GM =(COSGAM -SINGAM)*(COSGAM + SINGAM)                        MODELA
      GO TO 360                                                         MODELA
  350 PHI = PHID*RDI                                                    MODELA
      SINPHI = SIN(PHI)                                                 MODELA
      COSPHI = COS(PHI)                                                 MODELA
```

```
      CALPHA = APHR*RDI                                          MODELA
      ALPHA=CALPHA                                               FD14
  360 CALL BLGCON(JP1,JP2,JP3)                                   MODELA
      IF(ABS(APHR).GT.240..OR.ABS(VAR(2)).GT.7.) ISTART =6       IO
      IF(VAR(1).LT.0.) ISTART=6                                  IO
      IF(ABS(VAR(5)).GE.7.) ISTART=6                             IO
      CALL  DER                                                  MODELA
      CALL  SDER                                                 MODELA
      IF(JTYP.EQ.2) CALL ENVPAM                                  PHISZ
      RETURN                                                     MODELA
  999 CALL STPIT(4)                                              MODELA
      END                                                        MODELA
```

20 OCT 72 G 01-46

SUBRØUTINE
MØDELB

Subroutine MODELB

Entry MODEL

Purpose

Subroutine MODELB governs the computation of the adjoint coefficients (partial derivatives) during the backward adjoint integration.

Description

Subroutine MODELB performs a similar function to MODELA with the major exception that control bounding during the adjoint integration is determined by stored values of JP1 and JP3, rather than by inequality testing. MODELB is called from BNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALFMAX	α_{MAX}	I	Maximum angle of attack (DEG)	/ARCDAT/(	16)	INBVAD	M	ALFMAX
						MODELA	I	ALFMAX
						MODELB	I	ALFMAX
ALPHA	α	O	Angle of attack (RAD)	/AEC03 /(	3)	BEROCO	I	ALPHA
						BLGCON	M	ALPHA
						BL2	I	ALPHA
						FNTG	O	ALPHA
						MAMECO	I	ALPHA
						MODELA	M	ALPHA
						MODELB	O	ALPHA
						REU3	O	ALPHA
						VT	I	ALPHA
APHR	α	I	Angle of attack (DEG)	/AEC03 /(	2)	AGETB3	O	APHR
						AST3	M	APHR
						BEROCO	I	APHR
						BLGCON	O	APHR
						GUI3A	M	APHR
						MODELA	M	APHR
						MODELB	I	APHR
						MTX3A	O	APHR
						OUT	I	APHR
CALPHA		M	Constant value of angle-of-attack (RAD)	/GENF /(	552)	BL2	I	CALPHA
						MODELA	M	CALPHA
						MODELB	M	CALPHA
COSGAM	$\cos(\gamma)$	I	See symbol	/STATE3/(	687)	ACCEL	I	COSGAM
						BL4	I	COSGAM
						BL8	I	COSGAM
						DER3A	I	COSGAM
						EQUA3	O	COSGAM
						MODELA	I	COSGAM
						MODELB	I	COSGAM
						OUT	I	COSGAM
						PDBC	I	COSGAM
						PDY3A	I	COSGAM
COSPHI	$\cos\phi$	M	See symbol	/AEC03 /(	13)	ACCEL	I	COSPHI
						BL4	I	COSPHI
						GUI3A	M	COSPHI
						MODELA	M	COSPHI
						MODELB	M	COSPHI
						OUT	I	COSPHI
COSPSI	$\cos(\psi)$	I	See symbol	/STATE3/(	705)	BL4	I	COSPSI
						BL7	I	COSPSI
						BL8	I	COSPSI
						DER3A	I	COSPSI
						EQUA3	O	COSPSI
						MODELA	I	COSPSI
						MODELB	I	COSPSI
						PDBC	I	COSPSI
						PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol	/STATE3/(	707)	BL4	I	COSRHO
						BL7	I	COSRHO
						BL8	I	COSRHO
						DER3A	I	COSRHO
						EQUA3	M	COSRHO
						MODELA	I	COSRHO
						MODELB	I	COSRHO
						OUT	I	COSRHO
						PDBC	I	COSRHO
						PDY3A	I	COSRHO
COS2GM	$\cos(2\gamma)$	O	See symbol	/STATE3/(	757)	MODELA	O	COS2GM
						MODELB	O	COS2GM
COS2RO	$\cos(2\rho)$	O	See symbol	/STATE3/(	756)	BL4	I	COS2RO
						BL7	M	COS2RO
						BL8	M	COS2RO
						MODELA	O	COS2RO
						MODELB	O	COS2RO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
G	g	I	Gravitational attraction	(FT/SEC**2)	/GENF /(	301)	BL4	I	G
							BL7	I	G
							BL8	I	G
							DER3A	I	G
							EQUA3	M	G
							MODELA	I	G
							MODELB	I	G
							PDY3A	I	G
							SDER3	I	G
							SDINP	M	G
GMDOT	$\dot{\gamma}^*$	I	Pitch rate	(DEG/SEC)	/ARCDAT/(	15)	DER3A	I	GMDOT
							MODELA	I	GMDOT
							MODELB	I	GMDOT
							PROPB	I	GMDOT
							PROPIN	I	GMDOT
JGII		I	Control option		/XCODES/(	195)	ACCEL	I	JGII
							BNTG	O	JGII
							DER3A	I	JGII
							FNTG	M	JGII
							GUI3A	I	JGII
							MODELA	I	JGII
							MODELB	I	JGII
							MTX3A	I	JGII
							PDY3A	I	JGII
JP1		I	Option flag for first governing equation		/XCODES/(	217)	AGETB3	M	JP1
							AST3	M	JP1
							MODELA	M	JP1
							MODELB	I	JP1
							PROPB	O	JP1
							PROPIN	O	JP1
JP2		I	Option flag for second governing equation		/XCODES/(	218)	MODELA	I	JP2
							MODELB	I	JP2
							PROPB	O	JP2
							PROPIN	O	JP2
JP3		I	Option flag for third governing equation		/XCODES/(	219)	AGETB3	O	JP3
							AST3	M	JP3
							MODELA	M	JP3
							MODELB	I	JP3
							OUT	I	JP3
							PROPIN	O	JP3
OCORHO	$\omega \times$COSRHO	I	See symbol		/STATE3/(	708)	DER3A	I	OCORHO
							EQUA3	M	OCORHO
							MODELA	I	OCORHO
							MODELB	I	OCORHO
							PDBC	I	OCORHO
							PDY3A	I	OCORHO
OCORO2	$\omega \times$OCORHO	I	See symbol		/STATE3/(	709)	DER3A	I	OCORO2
							EQUA3	O	OCORO2
							MODELA	I	OCORO2
							MODELB	I	OCORO2
							PDY3A	I	OCORO2
OMGZ	ω	I	Earth rotation rate	(RAD/SEC)	/GLOBAL/(	3)	ABID3A	I	OMGZ
							CRASH	I	OMEGA
							DER3A	I	OMGZ
							EQUA3	I	OMGZ
							GEINP	I	OMGZ
							MODELA	I	OMGZ
							MODELB	I	OMGZ
							PDBC	I	OMGZ
							PDY3A	I	OMGZ
							SDINP	I	OMGZ
							TOPM	I	OMGZ
PHI	ϕ	M	Bank angle	(RAD)	/AECO3 /(	11)	GUI3A	M	PHIR
							MODELA	M	PHI
							MODELB	M	PHI
							OUT	I	PHI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
PHID	ϕ	I	Bank angle	(DEG)	/AEC03 /(	10)	AGETB3	O	PHID
							AST3	M	PHID
							GUI3A	M	PHID
							MODELA	M	PHID
							MODELB	I	PHID
							MTX3A	O	PHID
							OUT	I	PHID
R	R	I	Radial distance from earth center to vehicle	(FT)	/GENF /(	305)	BL4	I	R
							BL7	I	R
							BL8	I	R
							DER3A	I	R
							EQUA3	M	R
							MODELA	I	R
							MODELB	I	R
							PDBC	I	R
							PDY3A	I	R
							TRTOSZ	I	R
RDI		I	Angle to radian conversion, 01745329252		/DATA /(	3)	BLICO	I	RDI
							DER3A	I	RDI
							FNTG	I	RDI
							GUI3A	I	RDI
							MODELA	I	RDI
							MODELB	I	RDI
							PADSI	O	RDI
							PROPB	I	RDI
							PROPIN	I	RDI
							REU3	I	RDI
							SOINP	I	RDI
							SOMG	I	RDI
SINGAM	$\sin(\gamma)$	I	See symbol		/STATE3/(	688)	BL4	I	SINGAM
							BL7	I	SINGAM
							BL8	I	SINGAM
							DER3A	I	SINGAM
							EQUA3	O	SINGAM
							MODELA	I	SINGAM
							MODELB	I	SINGAM
							PDBC	I	SINGAM
							PDY3A	I	SINGAM
							SDER3	I	SINGAM
SINPHI	$\sin\phi$	M	See symbol		/AEC03 /(	12)	ACCEL	I	SINPHI
							BL4	I	SINPHI
							MODELA	M	SINPHI
							MODELB	M	SINPHI
							OUT	I	SINPHI
SINPSI	$\sin(\psi)$	I	See symbol		/STATE3/(	704)	BL4	I	SINPSI
							BL7	I	SINPSI
							BL8	I	SINPSI
							DER3A	I	SINPSI
							EQUA3	O	SINPSI
							MODELA	I	SINPSI
							MODELB	I	SINPSI
							PDBC	I	SINPSI
							PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol		/STATE3/(	706)	BL4	I	SINRHO
							BL7	I	SINRHO
							BL8	I	SINRHO
							DER3A	I	SINRHO
							EQUA3	O	SINRHO
							MODELA	I	SINRHO
							MODELB	I	SINRHO
							OUT	I	SINRHO
							PDBC	I	SINRHO
							PDY3A	I	SINRHO
SIN2RO	$\sin(2\rho)$	O	See symbol		/STATE3/(	755)	BL4	I	SIN2RO
							BL7	M	SIN2RO
							BL8	M	SIN2RO
							MODELA	O	SIN2RO
							MODELB	O	SIN2RO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBR	CODE	VAR
SQRT		F	Square root function	/SQRT	/(S)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTOE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAY02	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
V	v	I	Relative velocity (FT/SEC)	/STATE3	/(1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	D	KWOW
						YREF3	M	V
WORK		O	Working array, contains TOPEN1, TOPEN2, and PHIWT	/STS	/(3)	ADEQ3A	I	WORK
						FNTG	I	WORK
						MODELB	O	WORK
						MTX3A	I	WORK
						SDINP	M	WORK
						TEST	M	WORK
XKGAM	k_γ	M	Algebraic equation used in vertical rise and pitchover	/GENF	/(565)	BL4	I	XKG
						MODELA	M	XKGAM
						MODELB	M	XKGAM
XKPS	k_ψ	M	Algebraic equation used in vertical rise and pitchover	/GENF	/(566)	BL4	I	XKP
						MODELA	M	XKPS
						MODELA	I	XKPSI
						MODELB	M	XKPS
						MODELB	I	XKPSI
XKPSI	k_ψ	I	Algebraic equation used in vertical rise and pitchover	/GENF	/(566)	BL4	I	XKP
						MODELA	M	XKPS
						MODELA	I	XKPSI
						MODELB	M	XKPS
						MODELB	I	XKPSI

```
      SUBROUTINE  MODELB                                                MODELB
C       THIS  ROUTINE  COMPUTES  ADJOINT  COEFFICIENTS  ( D SUB Y . F )  MODELB
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR  (20),ITI      ,INTB     ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)          ,NCNST     ,NSB      ,NSAB     ,NICNB     ,  XCODES
     *I2OP     ,ICOP      ,IFAW      ,IFAR     ,IFB      ,IND       ,  XCODES
     *IOPEN    ,IPH       ,ISPH      ,ISST     ,IARC     ,ISTART    ,  XCODES
     *ITCT     ,ITER      ,IVAR      ,JK       ,JPS      ,JS        ,  XCODES
     *KOP      ,KPST      ,K         ,KST      ,NAD      ,NCASE     ,  XCODES
     *NCN      ,NEQB      ,NEQ       ,NOP      ,NPH      ,N         ,  XCODES
     *NST      ,IPST      ,IPRINT    ,ISTN     ,IPHN     ,ISTNB     ,  XCODES
     *IPHNB    ,IBLK1     ,IBLK2     ,ISTOP    ,ISTPP    ,L         ,  XCODES
     *IFOB     ,NB        ,LB        ,MB       ,NPHP     ,NPHB      ,  XCODES
     *NCTIN    ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/GENF/                                                      GENF
     *OMG(20)  ,OMGP(20,2),VARQ(9)   ,TOL(9)   ,SVAR(10) ,WDC(20)   ,  GENF
     *A(9,9)   ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9) ,DTP       ,  GENF
     *DTS      ,DT        ,G         ,DPSQ     ,Q        ,QS        ,  GENF
     *R        ,RE        ,MACH      ,PA       ,RO       ,CS        ,  GENF
     *VNU      ,PAR       ,ROR       ,CSR      ,VNR      ,SUMSQ     ,  GENF
     *SVSQ     ,TIMEPH    ,TIMES     ,TOP      ,TOS      ,TR(9)     ,  GENF
     *TST(20)  ,TPH   (20),DIS(20)   ,DIP(20)  ,T        ,W         ,  GENF
     *TLP1(20) ,TLS1  (20),DIP1(20)  ,DIS1(20) ,TIME     ,OMP       ,  GENF
     *TIMPR    ,LIFT      ,DRAG      ,TAX      ,TBURN    ,TBU(20)   ,  GENF
     *AE       ,FP        ,FPOLD     ,FPD      ,MACHR    ,MACHV     ,  GENF
     *QR       ,QV        ,FVAC      ,LIFTV    ,                       GENF
     *LIFTR    ,LIFTA     ,           DRAGV    ,DRAGR    ,DRAGA     ,  GENF
     *          LIFTM     ,DBR       ,DB       ,ISP      ,ISPF      ,  GENF
     *                     ULFT      ,ULFTV    ,ULFTR    ,ULFTA     ,  GENF
     *XMCG     ,XMCGV     ,XMCGR     ,XMCGA    ,XMCGM    ,CODAE     ,  GENF
     *CULFT    ,CT        ,CALPHA    ,COE      ,DELTAE   ,SID       ,  GENF
     *COD      ,SIDAE     ,XCG       ,ZCG      ,XJ                     GENF
      COMMON / GENF  /                                                  GENF
     *XJV      ,XJR       ,GH        ,GAMMAD   ,XKG      ,XKP       ,  GENF
     *FRATED   ,IRATED    ,                                             GENF
     *P1       ,P2        ,P3       ,XK1       ,XK2      ,XK3       ,  GENF
     *XK1T     ,XK2T      ,XK3T      ,XK1D     ,XK2D     ,XK3D      ,  GENF
     *XK1A     ,XK2A      ,XK3A      ,XK1V     ,XK2V     ,XK3V      ,  GENF
     *XK1G     ,XK2G      ,XK3G      ,XK1P     ,XK2P     ,XK3P      ,  GENF
     *XK1R     ,XK2R      ,XK3R      ,XK1O     ,XK2O     ,XK3O      ,  GENF
     *XK1U     ,XK2U      ,XK3U      ,XK1M     ,XK2M     ,XK3M      ,  GENF
     *PV       ,PG        ,PP        ,PR       ,PO       ,DPDY(3,8)    GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/DATA/                                                      DATA
     *PI       ,RAD       ,RDI       ,SC       ,UMF      ,TMPF      ,  DATA
     *FTNM     ,CAR       ,JOP1      ,JOP2     ,JOP3     ,JOP4         DATA
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)  ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)   ,SVY(10)   ,  STATE3D
     *XL(9,9)  ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM   ,SAVBP(15) ,  STATE3D
     *SINPSI   ,COSPSI    ,SINRHO    ,COSRHO   ,DCORHO   ,DCORO2    ,  STATE3D
     *SVBV   (9),OMEGA    ,OMEGA2    ,                                  STATE3D
     *VDV      ,GDV       ,RDV       ,MDV      ,PDV      ,ODV       ,  STATE3D
     *UDV      ,VDG       ,GDG       ,RDG      ,PDG      ,ODG       ,  STATE3D
     *UDG      ,VDR       ,GDR       ,MDR      ,PDR      ,ODR       ,  STATE3D
     *UDR      ,VDM       ,GDM       ,MDM      ,PDM      ,VOP       ,  STATE3D
     *GDP      ,PDP       ,ODP       ,UDP      ,VDO      ,GDO       ,  STATE3D
     *POD      ,UDD       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO   ,COS2RO    ,COS2GM                                      STATE3D
      COMMON/GLOBAL/                                                    GLOBAL
     *GR       ,ER        ,OMGZ      ,XLAMRF   ,YMURF    ,LUM           GLOBAL
     *,JJOP(10) ,IFATAL   ,NARC       ,NBRAN    ,NFARC    ,ID(4)        GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                                GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)     GLOBAL
     *,ITPSD ,KSOL      ,KGLOBL(8)                                      RETAP
      COMMON/AECO3/                                                     AECO3
     *APHO     ,APHR      ,ALPHA     ,VDA      ,GDA      ,PDA       ,  AECO3
     *SINA     ,COSA      ,PHIO      ,PHID     ,PHI      ,SINPHI    ,  AECO3
     *COSPHI   ,GOPH      ,POPH      ,XLAMA(9) ,XLAMP(9) ,CDO       ,  AECO3
     *CDOR     ,CLO       ,FK        ,XCGM     ,ZCGM     ,CLGM      ,  AECO3
     *CM       ,CMA       ,CMAM ,CMM ,CMO      ,CMOM     ,FKM       ,  AECO3
```

```
      *CLAM       ,CL         ,CLA        ,CLM        ,                     AECO3
      *CD         ,CDA        ,CDM                                          AECO3
       EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,    EQUV3
      *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2),   EQUV3
      *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),   EQUV3
      *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)              EQUV3
       REAL M,MU,MD                                                         EQUV3
       COMMON/ARCDAT/                                                       ARCDAT
      *SREF       ,EJ         ,XISP       ,TMULT      ,DTNC       ,DTPI   , ARCDAT
      *IATM       ,IMODE      ,JAER       ,JPRO       ,QMAX       ,GMAX   , ARCDAT
      *XLMAX      ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX      ,MAEA   , ARCDAT
      *MAEB       ,MAEC       ,MAED       ,MAEE       ,MAEF       ,MAEG   , ARCDAT
      *MT         ,MISP       ,MXCG       ,MZCG       ,MWDA       ,MWDB   , ARCDAT
      *MDB        ,XCGR       ,ZCGR       ,XE         ,ZE         ,XT     , ARCDAT
      *DREF       ,MCND       ,RHOB       ,QMULT      ,REMAX                ARCDAT
      * ,FRATE   , ARCD(9)                                                  RETAP
       DIMENSION  ARCDA(40)                                                 ARCDAT
       EQUIVALENCE(SREF,ARCDA)                                              ARCDAT
       COMMON/STS/                                                          STS
      *DPAY       ,PMIN       ,WORK (20),NWDS       ,IPC     (7),NITER    , STS
      *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)             STS
       EQUIVALENCE  (XKPS,XKP),(XKGAM,XKG)                                  MODELB
       EQUIVALENCE (XKPSI,XKP)                                              IO
       EQUIVALENCE(INQF,INEQFL(20))                                         APR27
       ENTRY  MODEL                                                         MODELB
       CALL AGETB                                                           P014
       INQF=JP3                                                             APR27
       ILAB(1)=0                                                            PHISZ
       IF(JP1 EQ.3) ILAB(1)=1                                               PHISZ
       CALL EQUA                                                            P014
       CALPHA =APHR *RDI                                                    MODELB
       ALPHA=CALPHA                                                         P014
       IF(JP3.NE.4)  GO TO 10                                               MODELB  10
       XKPS =SINRHO*SINPSI*(R*OCORO2+V*V*COSGAM*COSGAM/(R*COSRHO))       + MODELB
      1 2.*V*(OMGZ*SINRHO*COSGAM-OCORHO* COSPSI* SINGAM)                   MODELB
       XKGAM = OCORHO*(2 *V*SINPSI+ R*OMGZ *(COSRHO*COSGAM+ SINRHO*COSPSI MODELB
      1*SINGAM)) + COSGAM*(V*V/R- G)- V*RDI*GMDOT                          MODELB
       D= SQRT(XKGAM*XKGAM + XKPS*XKPS)                                     MODELB
       SINPHI = SIGN(1 ,XKGAM)* XKPSI/D                                     JULY28
       COSPHI= ABS(XKGAM)/ D                                                MODELB
       PHI=ATAN2(SINPHI,COSPHI)                                             P014
       SIN2RO = 2.*SINRHO*COSRHO                                            MODELB
       COS2RO = (COSRHO -SINRHO)*(COSRHO+ SINRHO)                           MODELB
       COS2GM =(COSGAM -SINGAM)*(COSGAM + SINGAM)                           MODELB
       GO  TO  20                                                           MODELB  20
C      I                  BANK  ANGLE  AND  SPECAIL TRIG  FUNCTIONS         MODELB
    10 PHI= PHID * RDI                                                      MODELB
       SINPHI = SIN(PHI)                                                    MODELB
       COSPHI = COS(PHI)                                                    MODELB
C      II   COMPUTE  DECISION VECTOR                                        MODELB
    20 CALL BLGCON(JP1,JP2,JP3)                                             P014
C      III   COMPUTE   PARTIALS OF DECISION VECTOR                          MODELB
       CALL  BLGCNV(JP1,JP2,JP3)                                            MODELB
C      III-A SET MULTIPLIER TO 0 IF NON-OPT CONTROL                         P014
       WORK(11)=1.                                                          P014
       IF(JP3.GT.2) WORK(11)=0.                                             P014
       IF(JGII.LE 10) WORK(11)=0.                                           APR
       IF(ALFMAX.EQ 0 ) GOTO 30                                             APR     30
       IF(ABS(APHR) GE.ALFMAX) WORK(11)=0.                                  P014
    30 CONTINUE                                                             APR
C      IV    COMPUTE   ADJOINT  COEFFICIENTS                                MODELB
       CALL  PDV                                                            MODELB
       RETURN                                                               MODELB
       END                                                                  MODELB
```

SUBRØUTINE
MØMJ

FUNCTION MOMJ

Entry MOMG

Purpose

MOMJ sets the internal stopping variable flag given the input variable code.

Description

MOMJ is called from SDINP during the boundary condition scan.

```
      FUNCTION  MOMJ( I)                                                MOMJ
C                                                                       COMN
C       SETS  STOPPING  VARIABLE FLAG  USING INPUT CODE                 COMN
C                                                                       COMN
      DIMENSION J(36)                                                   MOMJ
      DATA (J(N),N=1,36) /2,3,4,5,6,7,8,9,1,99,99,12,13,14,15,16,17,18, MOMJ
     1 19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36/           MOMJ
      ENTRY MOMG                                                        MOMJ
      L= IABS(I)                                                        MOMJ
      IF(L GT.36) GO TO 10                                              MOMJ   10
      K=  ISIGN(1,I)                                                    MOMJ
      IF(J(L).EQ.99) GO TO 10                                           MOMJ   10
      MOMJ =  K* J(L)                                                   MOMJ
      RETURN                                                            MOMJ
   10 WRITE(6,20) L                                                     MOMJ
      MOMJ=99                                                           MOMJ
   20 FORMAT(3X,I10,37H IS AN ILLEGAL STOPPING VARIABLE CODE)           MOMJ
      RETURN                                                            MOMJ
      END                                                               MOMJ
```

SUBRØUTINE MPSI

FUNCTION MPSI

Purpose

MPSI returns the internal constraint or payoff variable code given the input variable code.

Description

MPSI is called mainly from SDINP.

MPSI

```
      FUNCTION  MPSI(I)                                                 MOMJ
C                                                                       COMN
C        SETS   CONSTRAINT CODE FROM INPUT CODE                         COMN
      DIMENSION J(36)                                                   MOMJ
      DATA (J(N),N=1,36)/99,2,3,4,5,6,7,8,1,10,11,12,13,14,15,16,17,    MOMJ
     118,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36/         MOMJ
      IF(I.GT.36.OR.I.LT.1) GO TO 10                                    MOMJ
      JJ= J(I)                                                          MOMJ
      IF(JJ.EQ.99) GO TO 10                                             MOMJ
      MPSI= JJ                                                          MOMJ
      RETURN                                                            MOMJ
   10 WRITE(6,20)  I                                                    MOMJ
   20 FORMAT(3X,I3,30H IS AN ILLEGAL CONSTRAINT CODE   )                MOMJ
      MPSI= 99                                                          MOMJ
      RETURN                                                            MOMJ
      END                                                               MOMJ
```

20 OCT 72 6 01-46

512

SUBRØUTINE MTX3A

Subroutine MTX3A

Entry Points

MTX and MTX1

Purpose

MTX computes the TR vector during closed-loop control. MTX1 computes control and parameter changes during forward trajectory.

Description

The equations for calculating TR and the control and parameter changes are described in Sections 12. 1 and 15. 1 of Volume I. MTX3A entry points MTX and MTX1 are called from FNTG, and MODELA respectively.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	A	I	Control integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BSTO3	I	A
						MTX3A	I	A
						PAYO2	I	A
						SDINP	I	A
						TRAN3	I	A
APHO	α_{old}	I	Angle of attack from last nominal trajectory (DEG)	/AECO3 /(	1)	AST3	M	APHO
						FNTG	I	APHO
						MTX3A	I	APHO
						OUT	I	APHO
						PROPB	O	AEZRO
						PROPIN	O	AEZRO
APHR	α	O	Angle of attack (DEG)	/AECO3 /(	2)	AGETB3	O	APHR
						AST3	M	APHR
						BEROCO	I	APHR
						BLGCON	O	APHR
						GUI3A	M	APHR
						MODELA	M	APHR
						MODELB	I	APHR
						MTX3A	O	APHR
						OUT	I	APHR
B		M	Temp storage for a matrix also called B matrix	/GENF /(	208)	ADICB3	M	COTI
						MTX3A	M	B
						TRAN3	M	COTI
DALP	$\delta\alpha$	W	Angle of attack correction	/MTX3A /(*	)	MTX3A	W	DALP
DCON	$d\Psi_{,}$	I	Asked for correction in constraint misses and payoff vector	/GENF /(	289)	CON3	O	DCON
						MTX3A	I	DCON
						PAYO2	M	DCON
						TEST	M	DCON
						TOPM	I	DCON
						TRTOSZ	I	DCON
DPAR	δp	M	Adjustable parameter corrections	/PARAM /(	264)	ADJUST	I	DPAR
						MTX3A	M	DPAR
						TOPM	D	DPAR
DPHI	$\delta\phi$	W	Bank angle correction	/MTX3A /(*	)	MTX3A	W	DPHI
IND		I	Flag indicates whether on first nominal trajectory (IND=1)	/XCODES/(	141)	AST3	I	IND
						BGET3	I	IND
						FNTG	M	IND
						GUI3A	I	IND
						MTX3A	I	IND
						PROPIN	I	IND
JGII		I	Control option	/XCODES/(	195)	ACCEL	I	JGII
						BNTG	O	JGII
						DER3A	I	JGII
						FNTG	M	JGII
						GUI3A	I	JGII
						MODELA	I	JGII
						MODELB	I	JGII
						MTX3A	I	JGII
						PDY3A	I	JGII
NCN		I	Number of elements in $d\Psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BSTO3	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAYO2	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN

125

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BSTD3	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
NPA		I	Running count of number of adjustable parameters to be perturbed on remainder of trajectory	/PARAM /(	14)	ADJUST	M	NPAR
						FNTG	I	NPA
						MTX3A	I	NPA
						TOPM	D	NPA
NPARA		I	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	D	NPARA
PHID	ϕ	O	Bank angle (DEG)	/AECO3 /(	10)	AGETB3	O	PHID
						AST3	M	PHID
						GUI3A	M	PHID
						MODELA	M	PHID
						MODELB	I	PHID
						MTX3A	O	PHID
						OUT	I	PHID
PHIO	ϕ_{old}	I	Bank angle from last nominal trajectory	/AECO3 /(	9)	AST3	M	PHID
						MTX3A	I	PHID
RAD		I	Radian to angle conversion, 57.29577951	/DATA /(	2)	BEROCO	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADS1	D	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD
SPARB	$S^{\Psi}{}_{I}$	I	Matrix of adjustable parameter sensitivities. (Contains only elements corresponding to parameters yet to be adjusted)	/PARAM /(	144)	ADJUST	O	SPARB
						MTX3A	I	SPARB
						TOPM	D	SPARB
S2INV	[SS]	M	Parameter sensitivity contribution to A matrix	/PARAM /(	276)	ADJUST	O	S2INV
						MTX3A	M	S2INV
						PAYO2	M	S2INV
						TOPM	D	S2INV
TR		M	Vector modifier of impulse response function in control calculation	/GENF /(	322)	MTX3A	M	TR
						TRAN3	M	TR

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
VAR	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	V
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V
WORK		I	Working array, contains TOPEN1, TOPEN2, and PHIWT	/STS /(	3)	ADEQ3A	I	WORK
						FNTG	I	WORK
						MODELB	O	WORK
						MTX3A	I	WORK
						SDINP	M	WORK
						TEST	M	WORK
WTP	[Y]	I	Adjustable parameter diagonal weighting matrix order according to IPOINT.	/PARAM /(	132)	MTX3A	I	WTP
						PAYO2	I	WTP
						SDINP	O	WTP
						TOPM	O	WTP
XL	λ^{Ψ,Ω_J}	I	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
XLAMA	Λ^{Ψ,Ω_J}	I	Impulse response function column vector associated with angle of attack	/AECO3 /(	16)	ADEQ3A	M	XLAMA
						ADIC3A	O	XLAMA
						AST3	O	XLAMA
						BGET3	O	XLAMA
						BSTO3	M	XLAMA
						MTX3A	I	XLAMA
						TRAN3	M	XLAMA
XLAMP	Λ^{Ψ,Ω_J}	I	Impulse response function column vector associated with bank angle	/AECO3 /(	25)	ADEQ3A	M	XLAMP
						ADIC3A	O	XLAMP
						AST3	O	XLAMP
						BGET3	O	XLAMP
						BSTO3	M	XLAMP
						MTX3A	I	XLAMP
						TRAN3	M	XLAMP
YO	y_{old}	I	State vector of nominal trajectory	/STATE3/(	227)	AST3	O	YO
						MTX3A	I	YO

30 OCT 72 G 01-46

```
      SUBROUTINE MTX3A                                                  MTX3A
C                                                                       COMN
C       ENTRY MTX  COMPUTES  TR  VECTOR  DURING CLOSED  LOOP  CONTROL   COMN
C                                                                       COMN
C       ENTRY MTX1 COMPUTE  CONTROL  AND PARAMETER  CHANGES ON  TRAJECT. COMN
C                                                                       COMN
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                  STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VOO       ,GDD      , STATE3D
     *POO        ,UOO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                     STATE3D
      COMMON/AEC03/                                                     AEC03
     *APHO       ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      , AEC03
     *SINA       ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI   , AEC03
     *COSPHI     ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CD0      , AEC03
     *CDOM       ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM     , AEC03
     *CM         ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM      , AEC03
     *CLAM       ,CL        ,CLA       ,CLM       ,                      AEC03
     *CD         ,CDA       ,CDM                                        AEC03
      COMMON/STS/                                                       STS
     *DPAY       ,PMIN      ,WORK (20),NWDS       ,IPC    (7),NITER    , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)           STS
      DIMENSION RR(9),JJ(9),DELY(7),B(9,9)                              MTX3A
      EQUIVALENCE(DVARL(30),RR),(DVARL(40),DELY),(COTI,B),(DVARL(50),JJ) MTX3A
      COMMON/DATA/                                                      DATA
     *PI         ,RAD       ,RDI       ,SC        ,UMF       ,TMPF     , DATA
     *FTNM       ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4       DATA
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE    , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3        XCODES
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA     ,          DRAGV      ,DRAGR     ,DRAGA    , GENF
     *           LIFTM      ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                      ULFT       ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED     ,IRATED    ,                                            GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1Q      ,XK2Q      ,XK3Q     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
```

20 OCT 72 G.01-46

```
      *PV        ,PG        ,PP        ,PR        ,PO        ,DFDY(3,8)   GENF
       REAL  LIFTR , LIFT , (LIFTA,LIFTM ,              MACH, MACHR,      GENF
      * ISP, ISPF, MACHV,LIFTV , IRATED                                   FRAT
       DIMENSION TPH1(10),TST1(10)                                        GENF
       EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                 GENF
       COMMON/PARAM/                                                      PARAM
      *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),            PARAM
      *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                        PARAM
      *,DELP(9)                                                           PARAM
       ENTRY MTX                                                          MTX3A
C                                                                         COMN
C       I   COMPUTE  CHANGE IN STATE                                      COMN
       DO 10  I=1,NEQ                                                     MTX3A
    10 DELY(I) = VAR(I) - YO(I)                                           MTX3A
C                                                                         COMN
C       II   TEST  FOR  PARAMETER  SENSITIVITIES                          COMN
C            AND IF PRESENT  COMPUTE  CONTRIBUTION TO A MATRIX            COMN
       IF(NPA.EQ.0)  GO TO  40                                            MTX3A
       DO 30 I=1,NCN                                                      MTX3A
       DO 30 N=1,NCN                                                      MTX3A
       SS=0                                                               MTX3A
       DO 20 J=1,NPARA                                                    MTX3A
       SS= SS +  SPARB(I,J)*SPARB(N,J) / WTP(J)                           MTX3A
    20 CONTINUE                                                           MTX3A
       S2INV(I,N) = SS                                                    MTX3A
    30 CONTINUE                                                           MTX3A
C                                                                         CCMN
C       III    CHECK  FOR  ZERO  DIAGONAL  ELEMENT IN  A  AND S           COMN
C              MATRICES  AND  SET  FLAGS                                  COMN
    40 NO =0                                                              MTX3A
       DO 60 I=1,NCN                                                      MTX3A
       IF(A(I,I).EQ.0  AND S2INV(I,I).EQ.0.) GO TO 50                     MTX3A
       JJ(I)=1                                                            MTX3A
       NO =NO +1                                                          MTX3A
       GO TO  60                                                          MTX3A
    50 JJ(I) =0                                                           MTX3A
    60 CONTINUE                                                           MTX3A
C       IV   COMPRESS  A  MATRIX  AND  STORE  IN  B  WITH                 COMN
C              PARA. SENS. CONTRIBUTION                                   COMN
       IA =0                                                              MTX3A
       DO 80  I=1,NCN                                                     MTX3A
       IF(JJ(I).EQ.0) GO TO 80                                            MTX3A
       IA =IA+1                                                           MTX3A
       IB = IA -1                                                         MTX3A
       DO  70 J=I,NCN                                                     MTX3A
       IF( JJ(J).EQ.0) GO TO 70                                           MTX3A
       IB= IB+1                                                           MTX3A
       B(IA,IB) = A(I,J) - S2INV(I,J)                                     MTX3A
    70 CONTINUE                                                           MTX3A
    80 CONTINUE                                                           MTX3A
C       V   INVERT  A                                                     COMN
       IF(NO.GT.1) GO TO 90                                               MTX3A
       B(1,1)= 1./B(1,1)                                                  MTX3A
       GO TO 120                                                          MTX3A
    90 CONTINUE                                                           MTX3A
       CALL  SYMVRT( B,NO,IER)                                            MTX3A
       IF(IER.NE.0) GO TO 240                                             MTX3A
C      RESTORE MATRIX                                                     MTX3A
       DO 110 I=1,NO                                                      MTX3A
       DO 100 J=I,NO                                                      MTX3A
   100 B(I,J)= B(J,I)                                                     MTX3A
   110 CONTINUE                                                           MTX3A
C                                                                         COMN
C      VI  CALCULATE   CORRECTED  CONSTRAINT MISS VECTOR                  COMN
C             BASED  ON  CURRENT  EFFECT OF  CHANGE IN  STATE             COMN
   120 KK=0                                                               MTX3A
       DO 140 I=1,NCN                                                     MTX3A
       IF(JJ(I).EQ 0) GO TO 140                                           MTX3A
       KK=KK+1                                                            MTX3A
       RR(KK) =DCON(I)                                                    MTX3A
       DO 130  JP =1,NEQ                                                  MTX3A
   130 RR(KK) = RR(KK) - XL(JP,I)* DELY(JP)                               MTX3A
```

20 OCT 72 6 01-46

```
     140 CONTINUE                                                    MTX3A
C         VII     COMPUTE   TR   VECTOR   AND   RETURN               COMN
         DO 160 I=1,NO                                               MTX3A
         TR(I)=0.                                                    MTX3A
         DO 150 JP=1,NO                                              MTX3A
     150 TR(I)= TR(I) + B(I,JP)*RR(JP)                               MTX3A
     160 CONTINUE                                                    MTX3A
     170 CONTINUE                                                    MTX3A
         RETURN                                                      MTX3A
         ENTRY  MTX1                                                 MTX3A
         IF(IND.EQ.1)RETURN                                          P014
         KK=0                                                       .MTX3A
         DPHI =0.                                                    MTX3A
         DALP =0.                                                    MTX3A
C         VIII       ***                                             COMN
C     CALCULATE  CONTROL  CORRECTION                                 MTX3A
         DO  180 I=1,NCM                                             MTX3A
         IF(JJ(I).EQ.0) GO TO 180                                    MTX3A
         KK=KK+1                                                     MTX3A
         DALP = DALP + XLAMA(I)* TR(KK)                              MTX3A
         DPHI = DPHI + XLAMP(I)* TR(KK)                              MTX3A
       1  * WORK(10)                                                 MTX3A
     180 CONTINUE                                                    MTX3A
         IF(NPARA.EQ.0) GO  TO 210                                   MTX3A
         IF(NPA.EQ. 0)  GO  TO 210                                   MTX3A
C                                                                    COMN
C         IX      COMPUTE  PARAMETER  CORRECTIONS                    COMN
         DO 200  I=1,NPARA                                           MTX3A
         DPAR(I)= 0.                                                 MTX3A
         KK=0                                                        MTX3A
         DO 190  J=1,NCM                                             MTX3A
         IF(JJ(J).EQ.0) GO TO 190                                    MTX3A
         KK=KK+1                                                     MTX3A
         DPAR(I) = DPAR(I) - SPARB(KK,I)* TR(KK) /WTP(I)             MTX3A
     190 CONTINUE                                                    MTX3A
     200 CONTINUE                                                    MTX3A
     210 CONTINUE                                                    MTX3A
         IF(JGII.LE.10) RETURN                                       P014
         APHR = APHO  + DALP*RAD                                     MTX3A
         PHIO=   PHIO + DPHI * RAD                                   MTX3A
         IF(ABS( DPHI).LT..26) RETURN                                MTX3A
         PHIO=PHIO+SIGN(.26,DPHI) * RAD                              APR
         RETURN                                                      MTX3A
     220 CALL IPR(10HA SINGULAR, A, I, 81,0)                         MTX3A
     230 CALL  STPIT(4)                                              MTX3A
     240 CALL IPR(10HB SINGULAR, B, I, 81, 0)                        MTX3A
         GO TO 230                                                   MTX3A
         END                                                         MTX3A
```

20 OCT 72 G 01-46

526

SUBRØUTINE ØUT

Subroutine OUT

Entry OUTERR

Purpose

Subroutine OUT puts data into print array AP in preparation for printing a trajectory block.

Description

Subroutine OUT controls the computation of auxiliary print quantities (Section 8, Volume I) and checks print options. It loads all desired data into the print array AP and then calls subroutine PRINT which does the formatted printing, heading and labeling. Entry OUTERR is called whenever a control divergence occurs. This entry prints the message:

****CONTROL DIVERGENCE (SEE NEXT PRINT BLOCK)

And then continues with the remaining code in subroutine OUT to print a standard trajectory block. Subroutine OUT is called from FNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AINCL	i	I	Orbital inclination	(RAD)	/ORBIT /(	7)	OUT	I	AINCL
							PDBC	M	AINCL
ALT	h	I	Altitude		/STATE3/(	3)	EQUA3	I	ALT
							OUT	I	ALT
							PDBC	I	ALT
ANOMLY	ξ	I	True anomoly	(RAD)	/ORBIT /(	13)	OUT	I	ANOMLY
							PDBC	M	ANOMLY
APHO	α_{old}	I	Angle of attack from last nominal trajectory	(DEG)	/AECO3 /(	1)	AST3	M	APHO
							FNTG	I	APHO
							MTX3A	I	APHO
							OUT	I	APHO
							PROPB	O	AEZRO
							PROPIN	O	AEZRO
APHR	α	I	Angle of attack	(DEG)	/AECO3 /(	2)	AGETB3	O	APHR
							AST3	M	APHR
							BEROCO	I	APHR
							BLGCON	O	APHR
							GUI3A	M	APHR
							MODELA	M	APHR
							MODELB	I	APHR
							MTX3A	O	APHR
							OUT	I	APHR
APOGEE	R_a	I	Apogee radius	(FT)	/ORBIT /(	11)	OUT	I	APOGEE
							PDBC	O	APOGEE
ARGP	δ_p	I	Orbital argument of perigee	(RAD)	/ORBIT /(	8)	OUT	I	ARGP
							PDBC	M	ARGP
ASCNOD	Ω	I	Longitude of ascending node	(RAD)	/ORBIT /(	9)	OUT	I	ASCNOD
							PDBC	M	ASCNOD
CD	C_D	I	Drag coefficient		/AECO3 /(	52)	BEROCO	O	CD
							OUT	I	CD
							VT	I	CD
CL	C_L	I	Lift coefficient		/AECO3 /(	49)	BEROCO	M	CL
							OUT	I	CL
							VT	I	CL
COD	$\cos(\delta_E)$	I	See symbol		/GENF /(	556)	EL2	I	COD
							OUT	I	COD
							VT	M	COD
COSA	$\cos\alpha$	I	See symbol		/AECO3 /(	8)	ACCEL	I	COSA
							BL4	I	COSA
							BL6	I	COSA
							BL7	I	COSA
							BL8	I	COSA
							FH3	I	COSA
							OUT	I	COSA
							VT	M	COSA
COSGAM	$\cos(\gamma)$	I	See symbol		/STATE3/(	687)	ACCEL	I	COSGAM
							BL4	I	COSGAM
							BL8	I	COSGAM
							DER3A	I	COSGAM
							EQUA3	O	COSGAM
							MODELA	I	COSGAM
							MODELB	I	COSGAM
							OUT	I	COSGAM
							PDBC	I	COSGAM
							PDY3A	I	COSGAM
COSPHI	$\cos\phi$	I	See symbol		/AECO3 /(	13)	ACCEL	I	COSPHI
							BL4	I	COSPHI
							GUI3A	M	COSPHI
							MODELA	M	COSPHI
							MODELB	M	COSPHI
							OUT	I	COSPHI

524

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
COSRHO	$\cos(\rho)$	I	See symbol	/STATE3/(	707)	BL4	I	COSRHO
						BL7	I	COSRHO
						BL8	I	COSRHO
						DER3A	I	COSRHO
						EQUA3	M	COSRHO
						MODELA	I	COSRHO
						MODELB	I	COSRHO
						OUT	I	COSRHO
						PDBC	I	COSRHO
						PDY3A	I	COSRHO
CS	a	I	Speed of sound (FT/SEC)	/GENF /(	310)	EQUA3	M	CS
						OUT	I	CS
DB	D_b	I	Base drag (LBS)	/GENF /(	537)	ACCEL	I	DB
						BL4	I	DB
						BL6	I	DB
						BL7	I	DB
						BL8	I	DB
						EQUA3	I	DB
						FH3	I	DB
						OUT	I	DB
						SDER3	I	DB
						VT	I	DB
DELTAE	δ_E	I	Engine gimbal deflection angle (RAD)	/GENF /(	554)	BLGCON	M	DELTAE
						EL1	I	DELTAE
						OUT	I	DELTAE
						REU3	O	DELTAE
						VT	I	DELTAE
DPDY	$\partial w/\partial y$	I	Matrix of partials of in-plane control vector wrt state	/GENF /(	610)	ACCEL	I	DPDY
						BLGCON	I	DPDY
						OUT	I	DPDY
DRAG	D	I	Aerodynamic drag (LBS)	/GENF /(	497)	ACCEL	I	DRAG
						BL5	I	DRAG
						BL7	I	DRAG
						BL8	I	DRAG
						ENVPRM	I	DRAG
						FH3	I	DRAG
						OUT	I	DRAG
						PROPB	O	DRAG
						PROPIN	O	DRAG
						SDER3	I	DRAG
						VT	M	DRAG
ECC	e	I	Orbital eccentricity	/ORBIT /(	6)	OUT	I	ECC
						PDBC	M	ECC
ENERGY	E	I	Energy	/ORBIT /(	17)	OUT	I	ENERGY
						PDBC	O	ENERGY
FTNM		I	Feet to naut. mi. conversion, $1.645791629 \times 10^{-4}$	/DATA /(	7)	OUT	I	FTNM
						PADS1	D	FTNM
						TRTOSZ	I	FTNM
GAM	γ	I	Relative flight path angle (RAD)	/STATE3/(	2)	EQUA3	I	GAM
						GUI3A	I	GAM
						OUT	I	GAM
GAMI	γ_I	I	Inertial flight path angles (RAD)	/ORBIT /(	2)	OUT	I	GAMI
						PDBC	O	GAMI
GM	GM	I	Product of Newton's universal gravitational constant and the mass of the earth (FT^3/SEC^2)	/GLOBAL/(	67)	CRASH	I	GM
						OUT	I	GM
						PADS1	D	GM
						PDBC	I	GM

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
HMNTM	H	I	Momentum	/ORBIT /(	18)	OUT	I	HMNTM
						PDBC	M	HMNTM
HT	Q	I	Heating	/STATE3/(	8)	OUT	I	HT
HTD	$\dot{Q}$	I	Heating derivative	/STATE3/(	22)	DER3A	O	HTD
						OUT	I	HTD
						PDBC	I	HTD
						PDY3A	M	HTD
IATM		I	Atmosphere option flag	/ARCDAT/(	7)	EQUA3	I	IATM
						FXDAT	I	IATM
						OUT	I	IATM
						PDBC	I	IATM
						VT	I	IATM
IPFLG1		I	IPFLG1≠0 supresses print-out of velocity losses and inertial Euler angles.	/GLOBAL/(	69)	FNTG	I	IPFLG1
						OUT	I	IPFLG1
						PDBC	I	IPFLG1
						PRINT	I	IPFLG1
						TRTOSZ	O	IPFLG1
IPFLG3		I	IPFLG3≠0 supresses print-out of impact data.	/GLOBAL/(	71)	OUT	I	IPFLG3
						PRINT	I	IPFLG3
						TRTOSZ	O	IPFLG3
IPRINT		M	Print page counter initialization flag	/XCODES/(	168)	OUT	M	IPRINT
						TEST	I	IPRINT
						TOPM	O	IPRINT
ITCT		I	Iteration counter	/XCODES/(	148)	BNTG	I	ITCT
						OUT	I	ITCT
						TEST	M	ITCT
						TOPM	M	ITCT
ITER		I	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
						FNTG	I	ITER
						GETIT	I	ITER
						MODELA	I	ITER
						OUT	I	ITER
						PAYO2	M	ITER
						PROPIN	I	ITER
						TEST	M	ITER
						TOPM	M	ITER
JAER		I	Aerodynamic model option flag	/ARCDAT/(	9)	BEROCO	I	JAER
						EQUA3	I	JAER
						GEINP	I	JAER
						OUT	I	JAER
						PROPB	I	JAER
						PROPIN	I	JAER
						VT	I	JAER
JP3		I	Option flag for third governing equation	/XCODES/(	219)	AGETB3	O	JP3
						AST3	M	JP3
						MODELA	M	JP3
						MODELB	I	JP3
						OUT	I	JP3
						PROPIN	O	JP3

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
L		I	Integration traffic control flag L = 1 means evaluate derivatives = 2 check cut-off = 3 print or cut-off detected		/XCODES/(	177)	BNTG	M	L
							FNTG	M	L
							OUT	I	L
							RKTA3A	M	L
							RKTB3A	M	L
							SDINP	M	L
LIFT	L	I	Aerodynamic lift	(LBS)	/GENF /(	496)	ACCEL	I	LIFT
							BL4	I	LIFT
							BL5	I	LIFT
							BL6	I	LIFT
							ENVPRM	I	LIFT
							FH3	I	LIFT
							OUT	I	LIFT
							PROPB	O	LIFT
							PROPIN	O	LIFT
							VT	O	LIFT
M	m	I	Mass		/STATE3/(	4)	ACCEL	I	M
							BL4	I	M
							BL8	I	M
							EQUA3	I	M
							OUT	I	M
							SDER3	I	M
MACH	M	I	Mach number		/GENF /(	307)	BEROCO	I	MACH
							ENVPRM	I	MACH
							EQUA3	M	MACH
							OUT	I	MACH
MD	$\dot{m}$	I	MASS derivative		/STATE3/(	18)	DER3A	O	MD
							OUT	I	MD
MU	μ	I	Longitude		/STATE3/(	7)	OUT	I	MU
							PDBC	I	MU
NCN		I	Number of elements in dY		/XCODES/(	160)	ADEQ3A	I	NCN
							ADICB3	I	NCN
							ADIC3A	I	NCN
							ADID3A	I	NCN
							ADJUST	I	NCN
							AST3	M	NCN
							BNTG	I	NCN
							BST03	I	NCN
							MTX3A	I	NCN
							OUT	I	NCN
							PAY02	M	NCN
							TEST	M	NCN
							TOPM	I	NCN
							TRAN3	I	NCN
							TRTOSZ	I	NCN
NEQ		I	Number of integrated states		/XCODES/(	162)	ADICB3	I	NEQ
							ADIC3A	I	NEQ
							ADID3A	I	NEQ
							AGETB3	I	NEQ
							AST3	I	NEQ
							BGET3	I	NEQ
							BST03	I	NEQ
							MTX3A	I	NEQ
							OUT	I	NEQ
							REU3	I	NEQ
							SDER3	I	NEQ
							SDINP	M	NEQ
							TOPM	I	NEQ
							TRAN3	I	NEQ
							YREF3	I	NEQ
P	p_r	I	Semi-latus rectum	(FT)	/ORBIT /(	5)	OUT	I	P
							PDBC	M	P
PA	p_a	I	Atmospheric pressure	(PSF)	/GENF /(	308)	EQUA3	M	DZM
							FH2	I	PA
							IMPUL	I	PA
							OUT	I	PA
							PDBC	I	PA
							SDER3	I	PA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
PERGEE	R_p	I	Perigee radius	(FT)	/ORBIT /(	12)	OUT	I	PERGEE
							PDBC	O	PERGEE
PHI	ϕ	I	Bank angle	(RAD)	/AECO3 /(	11)	GUI3A	M	PHIR
							MODELA	M	PHI
							MODELB	M	PHI
							OUT	I	PHI
PHID	ϕ	I	Bank angle	(DEG)	/AECO3 /(	10)	AGETB3	O	PHID
							AST3	M	PHID
							GUI3A	M	PHID
							MODELA	M	PHID
							MODELB	I	PHID
							MTX3A	O	PHID
							OUT	I	PHID
PI	π	I	Constant 3.141592653		/DATA /(	1)	OUT	I	PI
							PADS1	D	PI
PSI	ψ	I	Azimuth		/STATE3/(	5)	EQUA3	I	PSI
							GUI3A	I	PSI
							OUT	I	PSI
PSII	ψ_I	I	Inertial azimuth	(RAD)	/ORBIT /(	3)	OUT	I	PSII
							PDBC	M	PSII
P1		I	First element in in-plane control vector. Corresponds to thrust		/GENF /(	569)	BLGCON	M	P
							BLGCON	M	P1
							OUT	I	P1
Q	q	I	Dynamic pressure	(PSF)	/GENF /(	303)	ENVPRM	I	Q
							EQUA3	M	Q
							OUT	I	Q
							PDBC	I	Q
							VT	I	Q
RAD		I	Radian to angle conversion, 57.29577951		/DATA /(	2)	BEROCO	I	DEG
							BLGCON	I	RAD
							ENVPRM	I	RAD
							EQUA3	I	RAD
							FNTG	I	RAD
							GUI3A	I	RAD
							MODELA	I	RAD
							MTX3A	I	RAD
							OUT	I	RAD
							PADS1	D	RAD
							SDINP	I	RAD
							TRTOSZ	I	RAD
RE	R_{ey}	M	Unit reynolds number	(1/FT)	/GENF /(	306)	OUT	M	RE
							PDBC	M	RE
RHO	ρ	I	Latitude		/STATE3/(	6)	EQUA3	I	RHO
							OUT	I	RHO
RO	ρ_a	I	Atmospheric density	(SLUGS/FT**3)	/GENF /(	309)	BL7	I	RO
							BL8	I	RO
							DER3A	I	RO
							EQUA3	I	RO
							OUT	I	RO
							PDBC	I	RO
							PDY3A	I	RO
SCROSS	S_C	I	Cross range	(FT)	/ORBIT /(	149)	OUT	I	SCROSS
							PDBC	O	SCROSS
SDOWN	S_D	I	Down range	(FT)	/ORBIT /(	148)	OUT	I	SDOWN
							PDBC	O	SDOWN
SID	$\sin(\delta_E)$	I	See symbol		/GENF /(	555)	EL2	I	SID
							OUT	I	SID
							VT	M	SID

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SINA	$\sin\alpha$	I	See symbol		/AECO3 /(	7)	ACCEL	I	SINA
							BL4	I	SINA
							BL6	I	SINA
							BL7	I	SINA
							BL8	I	SINA
							FH3	I	SINA
							GUI3A	M	SINA
							OUT	I	SINA
							VT	M	SINA
SINPHI	$\sin\phi$	I	See symbol		/AECO3 /(	12)	ACCEL	I	SINPHI
							BL4	I	SINPHI
							MODELA	M	SINPHI
							MODELB	M	SINPHI
							OUT	I	SINPHI
SINRHO	$\sin(\rho)$	I	See symbol		/STATE3/(	706)	BL4	I	SINRHO
							BL7	I	SINRHO
							BL8	I	SINRHO
							DER3A	I	SINRHO
							EQUA3	O	SINRHO
							MODELA	I	SINRHO
							MODELB	I	SINRHO
							OUT	I	SINRHO
							PDBC	I	SINRHO
							PDY3A	I	SINRHO
SMIMAJ	a_s	I	Semi-major axis	(FT)	/ORBIT /(	10)	OUT	I	SMIMAJ
							PDBC	M	SMIMAJ
SQRT		F	Square root function		/SQRT /($	)	ANLATM	F	SQRT
							CRASH	F	SQRT
							DCTOE	F	SQRT
							DER3A	F	SQRT
							ENVPRM	F	SQRT
							HUNT	F	SQRT
							MODELA	F	SQRT
							MODELB	F	SQRT
							OPWELL	F	SQRT
							OUT	F	SQRT
							PAT63	F	SQRT
							PAY02	F	SQRT
							PDBC	F	SQRT
							PDY3A	F	SQRT
							STORE	F	SQRT
							SYMVRT	F	SQRT
							WTSCH	F	SQRT
STOT	S_T	I	Total range	(FT)	/ORBIT /(	158)	OUT	I	STOT
							PDBC	O	STOT
							TRTOSZ	I	STOT
T	T	I	Thrust	(LBS)	/GENF /(	411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
TIME	t	I	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
TIMEPH	τ_p	I	Phase time (SEC)	/GENF /(	318)	EQUA3	O	TIMEPH
						FNTG	M	TIMEPH
						GETIT	I	TIMEPH
						GUI3A	I	TIMEPH
						OUT	I	TIMEPH
TIMES	τ	I	Arc time (SEC)	/GENF /(	319)	AST3	I	TIMES
						EQUA3	O	TIMES
						FNTG	M	TIMES
						GETIT	I	TIMES
						OUT	I	TIMES
V	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
VAR	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REV3	M	VAR
						RKTA3A	M	V
						STP3	I	VAR
						TOPM	O	KWOW
						VREF3	M	V
VI	V_I	I	Inertial velocity (FT/SEC)	/ORBIT /(	1)	OUT	I	VI
						PDBC	I	ORBPRM
						PDBC	M	VI
VNU	μ_a	I	Atmospheric viscosity [dynamic] (SLUGS/FT/SEC)	/GENF /(	311)	OUT	I	VNU
						PDBC	I	VNU
W	W	I	Weight (LBS)	/GENF /(	412)	BL5	I	W
						ENVPRM	I	W
						EQUA3	M	W
						FH3	I	W
						OUT	I	W
						PDBC	I	W
						REV3	I	W
						TRTOSZ	I	W
XJ	J	I	Control blend factor	/GENF /(	560)	EL2	I	XJ
						EQUA3	I	XJ
						OUT	I	XJ
						VT	I	XJ
XK3		I	Third control vector governing equation value. Corresponds to error in algebraic equation involving α.	/GENF /(	574)	BL2	O	XK3
						BL3	O	XK3
						BL4	O	XK3
						BL5	O	XK3
						BL6	O	XK3
						BL7	O	XK3
						BL8	O	XK3
						MODELA	I	XK3
						OUT	I	XK3
XL	λ^{Ψ}, Ω_J	I	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
XMCG	M_{CG}	I	Aerodynamic moment about center of gravity (FT-LBS)	/GENF /(	544)	EL2	I	XMCG
						OUT	I	XMCG
						VT	M	XMCG
XMUI	μ_I	I	Inertial longitude (RAD)	/ORBIT /(	4)	OUT	I	XMUI
						PDBC	M	XMUI

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
UN06		O	File of all output data	/.UN06./(*	)	BLICO	O	UN06.
						BNORYC	O	.UN06.
						CRASH	O	UN06.
						FRENCH	O	.UN06.
						FXDAT	O	UN06.
						GEINP	O	.UN06.
						HUNT	O	.UN06.
						INEDIT	O	.UN06.
						ITER8	O	UN06.
						MODELA	O	.UN06.
						MOMJ	O	.UN06.
						MPSI	O	UN06.
						OUT	O	.UN06.
						PAYO2	O	.UN06.
						PRINT	O	.UN06.
						PRINTV	O	.UN06.
						PRINTW	O	UN06.
						PRITEQ	O	.UN06.
						PRITVA	O	UN06.
						PROPIN	O	.UN06.
						PROTHR	O	UN06.
						PRWTSM	O	UN06.
						RANGE	O	.UN06
						S	O	.UN06.
						SDINP	O	.UN06.
						SIZE	O	.UN06.
						SIZIN	O	.UN06.
						SIZOUT	O	.UN06.
						SOLVE	O	.UN06.
						SPLICO	O	UN06.
						SPLIZ	O	UN06.
						SPLYNE	O	UN06.
						SSSP	O	.UN06.
						STAU	O	.UN06.
						STPIT	O	.UN06.
						SUMOUT	O	UN06.
						TABIN	O	.UN06.
						TEST	O	.UN06.
						VEHDF	O	.UN06.
						WTSCH	O	.UN06.
						WTVOL	O	UN06.

```
      SUBROUTINE OUT                                                    OUT3B
      COMMON/GENF/                                                      GENF
     *DMG(20)    ,DMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WOC(20)   , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *CDD        ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED   ,IRATED      ,                                           GENF
     *P1         ,P2        ,P3      ,XK1         ,XK2       ,XK3       , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V'     ,XK2V      ,XK3V      , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV         ,PS        ,PP        ,PR        ,PO        ,OPOY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,               MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10)   , STATE3D
     *XL(9,9)    ,YDP(20,9),YD5 (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    , STATE3D
     *SVBV    (9),OMEGA     ,OMEGA2    ,                                STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO        ,UDD       ,HTDV      ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                     STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,  EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2), EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD), EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)            EQUV3
      REAL M,MU,MD                                                      EQUV3
      COMMON/AECO3/                                                     AECO3
     *APHO       ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA       , AECO3
     *SINA       ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI    , AECO3
     *COSPHI     ,GDPH    . ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO       , AECO3
     *CDOM       ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM      , AECO3
     *CM         ,CMA       ,CMAM ,CMM ,CMG       ,CMOM      ,FKM       , AECO3
     *CLAM       ,CL        ,CLA       ,CLM       ,                       AECO3
     *CD         ,CDA       ,CDM                                        AECO3
      COMMON/ ORBIT/  VI,      GAMI,     PSII,      XMUI,        P,     ORBIT
     *     ECC,     AINCL,     ARGP,   ASCNOD,   SMIMAJ,   APOGEE,      ORBIT
     *  PERGEE,   ANOMLY,      CAPX,     CAPY,    ASYMP,   ENERGY,      ORBIT
     *   HMNTM,    DVIDV,    DVIDG,                                     ORBIT
     *  DVIDH ,    DVIDM,   DVIDPS,   DVIDRO,   DVIDMU,    DGIDV,       ORBIT
     *  DGIDG ,    DGIDH,   DGIDM ,   DGIDPS,   DGIDRO,    DGIDMU,      ORBIT
     *  DPIDV ,    DPIDG,   DPIDH ,   DPIDM ,   DPIDPS,    DPIDRO,      ORBIT
     *  DPIDMU,    DMIDV,   DMIDG ,   DMIDH ,   DMIDM ,    DMIDPS,      ORBIT
     *  DMIDRO,    DMIDMU,  DPDV  ,   DPDG  ,   DPDH  ,    DPDM  ,      ORBIT
     *  DPDPS ,    DPDRO ,  DPDMU ,   DECDV ,   DECDG ,    DECDH ,      ORBIT
     *  DECDM ,    DECDPS,  DECDRO,   DECDMU,   DIDV  ,    DIDG  ,      ORBIT
     *  DIDH  ,    DIDM  , DIDPS  ,  DIDRO  ,  DIDMU  ,   DBEDV  ,      ORBIT
     *  DBEDG ,    DBEDH ,  DBEDM ,   DBEDPS,   DBEDRO,    DBEDMU,      ORBIT
     *  DNODV ,    DNODG ,  DNODH ,   DNODM ,   DNODPS,    DNODRO,      ORBIT
     *  DNODMU,    DSMDV,    DSMDG,    DSMDH,    DSMDM,   DSMDPS        ORBIT
```

```
      COMMON/ORBIT/                                                      ORBIT
     *  OSMDRO,    OSMDMU,     DAPDV,     DAPDG,     DAPDH,    DAPDM ,   ORBIT
     *  DAPDPS,    DAPDRO,     DAPDMU,    DPEDV,     DPEDG,    DPEDH ,   ORBIT
     *DPEDM      ,DPEDPS     ,DPEDRO     ,DPEDMU     ,DANDV     ,DANDG   , ORBIT
     *  DANDH ,    DANDM ,     DANDPS,    DANDRO,    DANDMU,   DCXDV ,   ORBIT
     *  DCXDG ,    DCXDH ,     DCXDM ,    DCXDPS,    DCXDRO,   DCXDMU,   ORBIT
     *  DCYDV ,    DCYDG ,     DCYDH ,    DCYDM ,    DCYDPS,   DCYDRO,   ORBIT
     *  DCYDMU,    DASDV ,     DASDG ,    DASDH ,    DASDM ,   DASDPS,   ORBIT
     *  DASDRO,    DASDMU,     DENDV ,    DENDG ,    DENDH ,   DENDM ,   ORBIT
     *  DENDPS,    DENDRO,     DENDMU,    DMODV ,    DMODG ,   DMODH ,   ORBIT
     *  DMODM ,    DMODPS,     DMODRO,    DMODMU                         ORBIT
      DIMENSION ORBPRM(18),PPO(7,18)                                     ORBIT
      EQUIVALENCE (VI,ORBPRM),(DVIDV,PPO)                                ORBIT
      COMMON/ORBIT/ YMXRF,SNXLMR,CSXLMR,SDOWN,SCROSS,TD,TC               ORBIT
     *,SNPSR     ,CSPSR      ,SNGI       ,CSGI       ,SPSII     ,CPSII   , ORBIT
     * STOT      ,CSI        ,SNI        ,SNGNU      ,CSANO     ,COSDMU  , ORBIT
     * SINDMU    ,THT        ,WTFUEL                                     ORBIT
      COMMON/GLOBAL/                                                     GLOBAL
     *GR         ,ER         ,OMGZ       ,XLAMRF     ,YMURF     ,LUM       GLOBAL
     *,JJOP(10)  ,IFATAL     ,NARC       ,NBRAN      ,NFARC     ,ID(4)     GLOBAL
     *,KTAB(20),ITAB(20)    ,SIG     ,MAXTAB                               GLOBAL
     *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)       GLOBAL
     *,ITPSO ,KSOL      ,KGLOBL(8)                                         RETAP
      COMMON/DATA/                                                         DATA
     *PI         ,RAD        ,RDI        ,SC         ,UMF       ,TMPF    , DATA
     *FTNM       ,CAR        ,JOP1       ,JOP2       ,JOP3      ,JOP4      DATA
      COMMON /XCODES/                                                      XCODES
     *ITQ    (9),ICOR   (20),ITI         ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)             ,NCNST      ,NSB        ,NSAB      ,NICNB   , XCODES
     *I20P       ,ICOP       ,IFAW       ,IFAR       ,IFB       ,IND     , XCODES
     *IOPEN      ,IPH        ,ISPH       ,ISST       ,IARC      ,ISTART  , XCODES
     *ITCT       ,ITER       ,IVAR       ,JK         ,JPS       ,JS      , XCODES
     *KOP        ,KPST       ,K          ,KST        ,NAD       ,NCASE   , XCODES
     *NCN        ,NEQB       ,NEQ        ,NOP        ,NPH       ,N       , XCODES
     *NST        ,IPST       ,IPRINT     ,ISTN       ,IPHN      ,ISTNB   , XCODES
     *IPHNB      ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP     ,L       , XCODES
     *IFOB       ,NB         ,LB         ,MB         ,NPHP      ,NPHB    , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/ARCDAT/                                                       ARCDAT
     *SREF       ,EJ         ,XISP       ,TMULT      ,DTNC      ,DTPI    , ARCDAT
     *IATM       ,IMODE      ,JAER       ,JPRD       ,QMAX      ,GMAX    , ARCDAT
     *XLMAX      ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX     ,MAEA    , ARCDAT
     *MAEB       ,MAEC       ,MAED       ,MAEE       ,MAEF      ,MAEG    , ARCDAT
     *MT         ,MISP       ,MXCG       ,MZCG       ,MWDA      ,MWDB    , ARCDAT
     *MDB        ,XCGR       ,ZCGR       ,XE         ,ZE        ,XT      , ARCDAT
     *DREF       ,MCND       ,RHOB       ,QMULT      ,REMAX                ARCDAT
     * ,FRATE  ,  ARCD(9)                                                  RETAP
      DIMENSION  ARCDA(40)                                                 ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                              ARCDAT
      COMMON / PRINT/ AP(100)                                              OUT3B
      DATA AP / 100* 0377777777777777777777/                               P014
    1 CONTINUE                                                             P014
      GAMD= GAM*RAD                                                        IO
      PSIDG=PSI*RAD                                                        IO
      AP(1)= TIME                                                          OUT3B
      AP(2)= ALT                                                           OUT3B
      AP(3)= V                                                             OUT3B
      AP(4)=GAMD                                                           APR
      AP(5)=PSIDG                                                          APR
      AP(6)= MU*RAD                                                        OUT3B
      AP(7)= RHO*RAD                                                       OUT3B
      AP(8)= TIMES                                                         OUT3B
      AP(9)= W                                                             OUT3B
      MD=0                                                                 OUT3B
      CALL PDBC(-1,FF,FF,FF,0,MD )                                         OUT3B
      AP(10) = VI                                                          OUT3B
      AP(11) = GAMI*RAD                                                    OUT3B
      AP(12) = PSII*RAD                                                    OUT3B
      AP(13) = XMUI*RAD                                                    OUT3B
      AP(14) = SCROSS*FTNM                                                 P014
      AP(15)= TIMEPH                                                       OUT3B
      AP(16)= M                                                            OUT3B
      AP(17)=VAR(NEQ+1)                                                    SIZ
      AP(18)= HT                                                           OUT3B
      AP(19)= HTD                                                          OUT3B
```

```
      AP(20) = STOT *FTNM                                      P014
      AP(21) = SDOWN*FTNM                                      P014
      RE=0.                                                    FINI
      IF(IATM.NE.2) RE=V*RO/VNU                                FINI
      AP(22)= RE                                               OUT3B
      AP(23)= PA                                               OUT3B
      AP(24)= RO                                               OUT3B
      AP(25)= CS                                               OUT3B
      AP(26)=      MACH                                        APR
      AP(45)=JP3                                               APR
      AP(46)= XK3                                              APR
      AP(29) =APHR                                             OUT3B
       AP(30) =PHID                                            OUT3B
      SACHI = SINA*COSPHI                                      P014
      IF(ABS(SACHI).GT.1.) SACHI=SIGN(1.,SACHI)                P014
      SASHI=SINA*SINPHI                                        P014
      IF(ABS(SASHI).GT.1.) SASHI=SIGN(1.,SASHI)                P014
      AP(92)= GAMD + ASIN(SACHI)*RAD                           P014
      AP(93) = PSIDG + ASIN(SASHI)*RAD                         P014
      AP(31)= XJ                                               OUT3B
      AP(32)= Q                                                OUT3B
      AP(33)= XMCG                                             OUT3B
      AP(34)= LIFT                                             OUT3B
      AP(35)= DRAG                                             OUT3B
      AP(36)= T                                                OUT3B
      IF(MD.EQ.0.) GO TO 10                                    OUT3B
      AP(43)=-T/MD/GR                                          OUT3B
      GO TO 20                                                 OUT3B
   10 AP(43)=0.                                                OUT3B
   20 CONTINUE                                                 OUT3B
      AP(48)= DELTAE *RAD                                      OUT3B
      IF(JAER-2) 30,30,40                                      OUT3B
   30 AP(49)= (T + LIFT*SINA-DB - DRAG*COSA) / W               OUT3B
      AP(50)= ( LIFT*COSA+ DRAG*SINA)/ W                       OUT3B
      GO TO 50                                                 OUT3B
   40 AP(49)= (T*COD +LIFT*SINA-DB -DRAG*COSA) / W             OUT3B
      AP(50)=(LIFT*COSA+DRAG*SINA-T*SID)/W                     P014
   50 AP(51)=SQRT(AP(49)**2+AP(50)**2)                         P014
      AP(27)= CL                                               P014
      AP(28)= CD                                               P014
      AP(37)= XL(1,NCM)                                        AAA
      AP(38)= XL(2,NCM)                                        AAA
      AP(39)= XL(5,NCM)                                        AAA
      AP(40)= XL(3,NCM)                                        AAA
      AP(41)= XL(6,NCM)                                        AAA
      AP(42)= XL(7,NCM)                                        AAA
      AP(44)= XL(4,NCM)                                        P014
      CALL  COOROS(COSRHO,SINRHO)                              P014
      IF(IPFLG1.NE.0) GO TO  60                                OUT3B
      AP(62)= SMIMAJ *FTNM                                     P014
      AP(63)=ECC                                               OUT3B
      AP(64)=AINCL * RAD                                       OUT3B
      AP(65)= ASCNOD* RAD                                      OUT3B
      AP(66)= ARGP * RAD                                       OUT3B
      AP(67)= APOGEE * FTKM                                    P014
      AP(68)= PERGEE * FTNM                                    P014
      AP(69)= ANOMLY *RAD                                      OUT3B
      AP(70)= SMIMAJ *2.*PI*SQRT(     SMIMAJ/GM)               OUT3B
      AP(71)=ENERGY                                            OUT3B
      AP(72)=HMNTM                                             OUT3B
      AP(73) = P*FTNM                                          P014
      AP(74)=HMNTM /APOGEE                                     OUT3B
      AP(75)=HMNTM / PERGEE                                    OUT3B
   60 CONTINUE                                                 OUT3B
      IF(IPFLG3. NE. 0) GO TO 70                               OUT3B
      IF(ITER.NE.3) GO TO 90                                   P014
      IF(V.LT.300..OR.ABS(COSGAM).LT..001) GO TO 70            FINI
      CALL CRASH                                               OUT3B
   70 CONTINUE                                                 OUT3B
      IF(ITER.LT.3) GO TO 90                                   P014
      IF(IPFLG1.NE.0) GO  TO 90                                OUT3B
      AP(55)=VAR(NEQ+2)                                        SIZ
```

20 OCT 72 G.01-46

```
      AP(56)=VAR(NEQ+4)                                              SIZ
      AP(57)= VAR(NEQ+5)                                             SIZ
      AP(58)= VAR(NEQ+3)                                             SIZ
   90 CONTINUE                                                       OUT38
      CALL PRINT(ITCT,ITER/3,0,IPRINT)                               OUT38
      IPRINT =0                                                      P014
      RETURN                                                         OUT38
      ENTRY OUTERR                                                   P014
      WRITE (6,100)                                                  P014
  100 FORMAT (49H ****  CONTROL DIVERGANCE (SEE NEXT PRINT BLOCK) )  P014
      CALL PDUMP(APHO,PHI,1,L,JP3,2,P1,DPDY(3,7),1)                  P014
      GO TO 1                                                        P014   1
      END                                                            OUT38
```

20 OCT 72 6 01-46

SUBRØUTINE PAT63

Subroutine PAT63

Purpose

Subroutine PAT63 computes atmosphere properties and derivatives for the 1963 Patrick Air Force Base atmosphere model.

Description

The reference for this atmosphere model is noted in Vol I, Section 3.3.2. This model has some programming limitation that should be noted. Above about 360,000 feet, the mathematical representation is very poor; therefore, the properties are automatically fixed at that point and partials are set to zero. Below sea level, a similar procedure is employed in the subroutine.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SC		I	Constant in Sutherlands equation, 198	/DATA	/(4)	PADS1 PAT63	D I	SC SC
SQRT		F	Square root function	/SQRT	/( )	ANLATM CRASH DCTOE DER3A ENVPRM HUNT MODELA MODELB OPWELL OUT PAT63 PAY02 PDBC PDY3A STORE SYMVRT WTSCH	F F F F F F F F F F F F F F F F F	SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT SQRT
TF		I	Constant in Sutherlands equation, 392	/DATA	/(6)	PADS1 PAT63	D I	TMPF TF
UMUF		I	Constant in Sutherlands equation, $.301\times10^{-6}$	/DATA	/(5)	PADS1 PAT63	D I	UMF UMUF

544


```
      SUBROUTINE PAT63( Q,PV,MGO)                                       PAT63I
      DIMENSION PV(8)                                                   PAT63I
      DATA  R0,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13 /             PAT63I
     1          8.3680902734182425, 3.6752156174349436      ,           PAT63I
     2          .5487134541479621 5, 1.4678068127507852  .  ,           PAT63I
     3         -.61463378717574668, -1.5399754331174728     ,           PAT63I
     4          82668415705291311,  1 2852428316497672      ,           PAT63I
     5         -.57234501173756826, -.62238012815282961     ,           PAT63I
     6          .17229840038525958,   14809576798626313     ,           PAT63I
     7         -.018335310293538199 ,-.01349827673004048    /           PAT63I
      DATA  S0,S1,S2,S3,S4,S5,S6,S7,S8,S9,S10,S11,S12,S13 /             PAT63I
     1          8.497018499534999    ,4 168051278711044    ,            PAT63I
     2          1.001136979720595    ,  3756892095830644   ,            PAT63I
     3         -1.287997408735040    , 0 014970860707497 03,            PAT63I
     4          1 674006844535595    , .1300346805855508 ,              PAT63I
     5         -1.176664525367179    , -.248063347106903 ,              PAT63I
     6          .3640551544325923    ,  1062741660382213 ,              PAT63I
     7         -.03986181926386985 , -.013645479194274 45 /             PAT63I
      DATA  T0,T1,T2,T3,T4,T5,T6,T7,T8,T9,T10,T11,T12,T13 /             PAT63I
     1              -.10248437640715577  , -.20389894971168512 ,        PAT63I
     2              -.020399380671279852 , .33088492304959274   ,       PAT63I
     3              -.55473185906456939  , -.55482818087304384  ,       PAT63I
     4               .98595655569038398  , .67846024779392942   ,       PAT63I
     5              - 66677674378491896  ,- 40978650717250266   ,       PAT63I
     6               .19688591214668486  , .11368136922321633   ,       PAT63I
     7            -0 020901390466879155 ,-0 011595950191818112 /        PAT63I
      DATA  U0,U1,U2,U3,U4,U5,U6,U7,U8,U9,U10,U11,U12 /                 PAT63I
     13 67521561743494  ,  1.09742690029592 ,  4.40342043825234 ,       PAT63I
     2-2 45853514870299, -7.69987716558737  ,  4.96010494231746 ,       PAT63I
     38.99669982154836  , -4.57876009390054 , -5.60142115337547 ,       PAT63I
     41 72298400385259  ,  1 62905349784890 , - 22002372352245 8 ,      PAT63I
     5-.175477597490526/                                                PAT63I
      DATA  V0,V1,V2,V3,V4,V5,V6,V7,V8,V9,V10,V11,V12 /                 PAT63I
     14 16805127871103  ,  2.00227395944118 ,  1 12706762874920 ,       PAT63I
     2-5.15198963494015, 0.07485430353748 52,  10.0440410672136 ,       PAT63I
     3.91024276409885 4 , -9.41331620293744 , -2.23257012396212 ,       PAT63I
     43.64055154432592  ,  1.16901582642043 , -.47834183116643 7 ,      PAT63I
     5-.17739122952557 /                                                PAT63I
      DATA  W0,W1,W2,W3,W4,W5,W6,W7,W8,W9,W10,W11,W12/                  PAT63I
     1- 203898949711685, -.040798761342559 8, .99265476914877 8    ,    PAT63I
     2-2 21892743625828, -2.77414090436523  , 5 91573933414227     ,    PAT63I
     34.74922173455749  , -5.33421395027935  ,-3.68807856455253    ,    PAT63I
     41 96885912146685  ,  1.25049506145537  ,-0.250816685602551   ,    PAT63I
     5-0.150747352493635/                                               PAT63I
      DATA  RHO, P0,A0 /.00229645   , 2124.0827,1137.98   /             PAT63I
      DATA  X0,X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,X11 /                     PAT63I
     *1.09742690829592  ,  8.80684087650468  , -7.37560544610898 ,      PAT63I
     *-30 7995086623495, 24.800524711587 29 , 53 98019892929005 ,       PAT63I
     *-32 0513206573037,-44.8113692270037 5 , 15.50685603467332 ,       PAT63I
     *16.29053447848889, -2.42026095874704  , -2.10573116988631 /       PAT63I
      DATA  Z0,Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9,Z10,Z11 /                     PAT63I
     *-.040798761342559 8,  1.98530953829756 , -6 65678230877484 ,      PAT63I
     *-11 0965636174609 , 29.57869667071134 , 28 49533040734491 ,       PAT63I
     *-37 3394976519553 ,-29.50462851642021 , 17.71973209320163 ,       PAT63I
     *12.50495061455371 , -2.75898354162806 , -1.80896822992361 /       PAT63I
      COMMON / SPECO  / UMU2 ,RORR                                      PAT63I
      DATA COEF1,TFSC / 1.7818E-4,590./                                 PAT63I
      COMMON/GLOBAL/IDUMM(91),JP3                                       APR27
      DATA   CONST1, CONST2/                                            PAT63I
     * 4.161533415E-4, 8 323066830E-4/                                  PAT63I
      EQUIVALENCE (TF,TMPF),(UMUF,UMF)                                  PAT63I
      COMMON/DATA/                                                      DATA
     *PI          ,RAD         ,RDI         ,SC          ,UMF          ,TMPF        ,  DATA
     *FINM        ,CAR         ,JOP1        ,JOP2        ,JOP3         ,JOP4          DATA
      H=Q                                                               PAT63I
      IF(H.LT.0.) H=0.                                                  PAT63I
      IF(H GT 370000.) H=370000.                                        JULY28
      X= (H- 2 E5)*1.E-5                                                PAT63I
      PXRHO = R0 + X*(R1+ X*(R2 +X*(R3 +X*(R4 +X*(R5 +X*(R6 +X*(R7 +X*( PAT63I
     1  R8 +X*(R9 +X*(R10+X*(R11+X*(R12 +X*R13)))))))))))))             PAT63I
      PXP   = S0 + X*(S1+X *(S2 +X*(S3 +X* (S4 +X*(S5 +X*(S6 +X*(S7 +X*( PAT63I
     1  S8 +X*(S9 +X*(S10+X*(S11+X*(S12 +X*S13)))))))))))))             PAT63I
      PXA   = T0 + X*(T1+X *(T2 +X*(T3 +X* (T4 +X*(T5 +X*(T6 +X*(T7 +X*( PAT63I
     1  T8 +X*(T9 +X*(T10+X*(T11+X*(T12 +X*T13)))))))))))))             PAT63I
      PV(1)=  P0*EXP(-PXP)                                              PAT63I
```

20 OCT 72 S.01-46

545

```
      PV(2)=  RHO*EXP(-PXRHO)                                           PAT63I
      PV(3)=  AO *EXP( PXA)                                             PAT63I
      TM = PV(3)**2 * CONST1                                            PAT63I
      FT  = (TM/TF)* SQRT(TM/TF)                                        PAT63I
      TSC =  TM + SC                                                    PAT63I
      UMU =  COEF1 * FT/TSC                                             PAT63I
      PV(4)=UMU                                                         PAT63I
      IF(Q GT 370000.) GO TO 30                                         JULY28
      IF(MGO EQ 1.AND.JP3.LT 7) RETURN                                  PAT63I
      DXRHO = U0 +X*(U1 +X*(U2+X*(U3+X*(U4+X*(U5+X*(U6 +X*(U7 +X*(U8+    PAT63I
     1  X*(U9 +X*(U10 +X*(U11+X*U12))))))))))))                         PAT63I
      DXP   = V0 + X*(V1 +X*(V2+X*(V3+X*(V4+X*(V5+X*(V6 +X*(V7 +X*(V8+   PAT63I
     1  X*(V9 +X*(V10 +X*(V11+X*V12))))))))))))                         PAT63I
      DXA   = W0 + X*(W1 +X*(W2+X*(W3+X*(W4+X*(W5+X*(W6 +X*(W7 +X*(W8+   PAT63I
     1  X*(W9 +X*(W10 +X*(W11+X*W12))))))))))))                         PAT63I
      PV(5) = -PV(1)*DXP  *1.E-5                                        PAT63I
      PV(6) = -PV(2)*DXRHO*1.E-5                                        PAT63I
      PV(7) =  PV(3)*DXA *1.E-5                                         PAT63I
      TM1 = PV(3)* PV(7) * CONST2                                       PAT63I
      GT   = TFSC/TSC                                                   PAT63I
      GT1  = -GT /TSC                                                   PAT63I
      FT1  = 1 5* FT/TM                                                 PAT63I
      UMU1 = UMUF*(FT* GT1 + GT* FT1)* TM1                              PAT63I
      PV(8)= UMU1                                                       PAT63I
      IF(Q.GE.0.) GO TO 5                                               PHISZ
      DO 3 III=5,8                                                      PHISZ
    3 PV(III)=0                                                         PHISZ
    5 CONTINUE                                                          PHISZ
      IF(MGO.EQ 1) RETURN                                               PAT63I
      IF(JP3 LT 7) RETURN                                               PAT63I
      IF(JP3-8)  20,20,10                                               PAT63I
   10 CONTINUE                                                          PAT63I
      QXA   = Z0 + X*(Z1 + X*(Z2 + X*(Z3 + X*(Z4 + X*(Z5 + X*(Z6 + X*(Z7 PAT63I
     * + X*(Z8 + X*(Z9 + X*(Z10 + X*Z11))))))))))                       PAT63I
      ARR =  1 E-10*PV(3)*(QXA +DXA*DXA)                                PAT63I
      TM2 = CONST2*(PV(3)*ARR + PV(7)*PV(7))                            PAT63I
      TM20= TM*TM                                                       PAT63I
      GT2  = 2 * GT/ TSC/TSC                                            PAT63I
      FT2  = 0 75* FT/TM20                                              PAT63I
      UMU2 = UMUF*((FT*GT1 + GT*FT1)* TM2 +                             PAT63I
     *        (FT*GT2+ 2 *FT1*GT1 + GT*FT2)* TM1* TM1)                  PAT63I
   20 CONTINUE                                                          PAT63I
      QXRHO = X0 + X*(X1 + X*(X2 + X*(X3 + X*(X4 + X*(X5 + X*(X6 + X*(X7 PAT63I
     * + X*(X8 + X*(X9 + X*(X10 + X*X11))))))))))                       PAT63I
      RORR = -1 E-10*PV(2)*(QXRHO -DXRHO*DXRHO)                         PAT63I
      RETURN                                                            PAT63I
   30 DO 40 I=5,8                                                       JULY28
   40 PV(I)=0.                                                          JULY28
      UMU2=0.                                                           JULY28
      RORR=0.                                                           JULY28
      RETURN                                                            JULY28
      END                                                               PAT63I
```

SUBRØUTINE PAYØ2

PAY02

Purpose

PAY02 computes the payoff improvement.

Description

The method of computing the payoff improvement is described in Sections 12.2 and 15.3 of Volume I.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	A	I	Control integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BSTO3	I	A
						MTX3A	I	A
						PAYO2	I	A
						SDINP	I	A
						TRAN3	I	A
B	B	W	B matrix	/PAYO2 /(*	)	PAYO2	W	B
DC	$d\Phi$	W	Predicted pay-off improvement	/PAYO2 /(*	)	PAYO2	W	DC
DCON	$d\Psi_{I}$	M	Asked for correction in constraint misses and payoff vector	/GENF /(	289)	CON3	O	DCON
						MTX3A	I	DCON
						PAYO2	M	DCON
						TEST	M	DCON
						TOPM	I	DCON
						TRTOSZ	I	DCON
DPAY	$d\Phi$	I	Initial payoff improvement	/STS /(	1)	PAYO2	I	DPAY
						SDINP	I	DPAY
						SDINP	O	IST
						SDINP	I	ST
						TEST	I	DPAY
						TOPM	O	IDPAY
DPSQ	$(dP)^2$	M	Metric of control and parameter changes [sd]	/GENF /(	302)	PAYO2	M	DPSQ
						TEST	M	PSISQ
						TRTOSZ	I	DPSQ
DSQ		W	Terminal constraint elements	/PAYO2 /(*	)	PAYO2	W	DSQ
IPASS		I	Sizing iteration counter	/SIZING/(	291)	GEINP	O	IPASS
						PADS1	M	IPASS
						PAYO2	I	IPASS
						SIZE	M	IPASS
						SIZIN	I	IPASS
						SSSP	M	IPASS
ITER		M	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
						FNTG	I	ITER
						GETIT	I	ITER
						MODELA	I	ITER
						OUT	I	ITER
						PAYO2	M	ITER
						PROPIN	I	ITER
						TEST	M	ITER
						TOPM	M	ITER
I2OP		M	First optimization pass flag sets $d\Phi$ = DPAY. Also used to indicate payoff degradation due to restoration of constraints	/XCODES/(	136)	PAYO2	M	I2OP
						TEST	M	I2OP
						TOPM	O	I2OP
JJ		W	Pointer array indicates zeros on A matrix diagonal	/PAYO2 /(*	)	PAYO2	W	JJ
JK		M	Integration routine flag tells which derivative evaluation in Runge-Kutta cycle	/XCODES/(	151)	ADIC3A	M	JK
						BNTG	I	JK
						MODELA	I	JK
						PAYO2	M	JK
						RKTA3A	M	J
						RKTB3A	M	J
NCC		W	Number of elements in $d\Psi$ at terminus excluding pay-off.	/PAYO2 /(*	)	PAYO2	W	NCC

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NCN		M	Number of elements in $d\psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADIC83	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BSTO3	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAYO2	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BSTO3	I	NCNST
						CON3	I	NCNST
						PAYO2	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST
NO		W	Number of non-zero A diagonal elements	/PAYO2 /(*	)	PAYO2	W	NO
NPARA		I	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	D	NPARA
PMIN		I	Minimum payoff improvement	/STS /(	2)	PAYO2	I	PMIN
						SDINP	I	PMIN
						TEST	I	PMIN
RAT		W	Ratio of current last pay-off improvement	/PAYO2 /(*	)	PAYO2	W	RAT
SIG		I	Payoff sign. SIG < 0 Payoff to be minimized, SIG > 0 Payoff to be maximized.	/GLOBAL/(	65)	PAYO2	I	SIG
						SDINP	M	SIG
						TEST	I	SIG
						TRAN3	I	SIG
SPARA	$S^{\dagger}{}_{I}$	I	Matrix of adjustable parameter sensitivities (including all parameters)	/PARAM /(	15)	ADJUST	I	SPARA
						PAYO2	I	SPARA
						STAU	M	SPARA
						TOPM	D	SPARA
SQRT		F	Square root function	/SQRT /(5	)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTOE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAYO2	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
SVDPSQ		I	Saved control matric	/SIZING/(	319)	PAYO2	I	SVDPSQ
						TRTOSZ	D	SVDPSQ

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBR	CODE	VAR
S2INV	[SS]	M	Parameter sensitivity contribution to A matrix	/PARAM /(	276)	ADJUST	O	S2INV
						MTX3A	M	S2INV
						PAY02	M	S2INV
						TOPM	O	S2INV
WTP	[Y]	I	Adjustable parameter diagonal weighting matrix order according to IPOINT.	/PARAM /(	132)	MTX3A	I	WTP
						PAY02	I	WTP
						SOINP	O	WTP
						TOPM	O	WTP
XM		W	Minor of B matrix	/PAY02 /(*	)	PAY02	W	XM
UN06.		O	File of all output data	/ UN06 /($	)	BLICO	O	.UN06.
						BNDRYC	O	.UN06.
						CRASH	O	.UN06.
						FRENCH	O	.UN06.
						FXDAT	O	.UN06.
						GEINP	O	.UN06.
						HUNT	O	.UN06.
						INEDIT	O	.UN06.
						ITER8	O	.UN06.
						MODELA	O	.UN06.
						MOMJ	O	.UN06.
						MPSI	O	.UN06.
						OUT	O	.UN06.
						PAY02	O	.UN06.
						PRINT	O	.UN06.
						PRINTV	O	.UN06.
						PRINTW	O	.UN06.
						PRITEQ	O	.UN06.
						PRITVA	O	.UN06.
						PROPIN	O	.UN06.
						PROTHR	O	.UN06.
						PRWTSM	O	.UN06.
						RANGE	O	UN06.
						S	O	.UN06.
						SOINP	O	.UN06.
						SIZE	O	.UN06.
						SIZIN	O	.UN06.
						SIZOUT	O	.UN06.
						SOLVE	O	UN06.
						SPLICO	O	.UN06.
						SPLIZ	O	.UN06.
						SPLYNE	O	UN06.
						SSSP	O	.UN06.
						STAU	O	.UN06.
						STPIT	O	.UN06.
						SUMOUT	O	.UN06.
						TABIN	O	.UN06.
						TEST	O	UN06.
						VEHDF	O	.UN06.
						WTSCH	O	.UN06.
						WTVOL	O	.UN06.

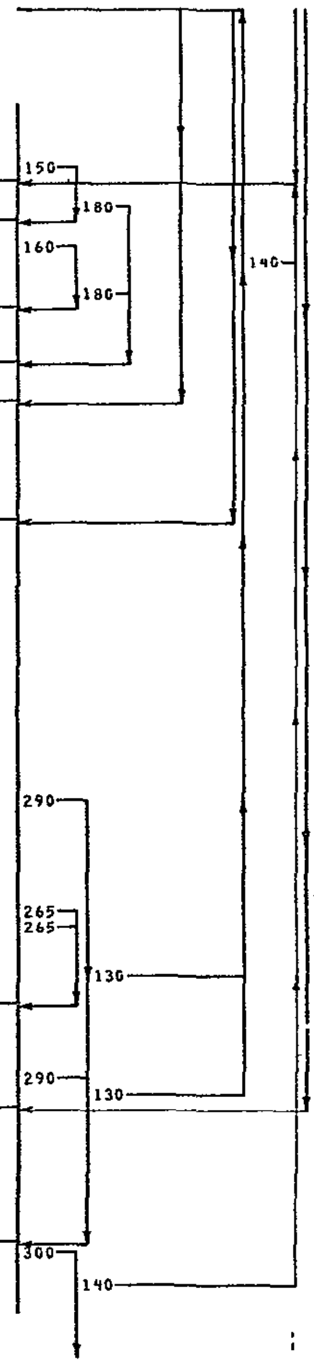
```
      PROGRAM PAYO2                                                     PAYO2
C                                                                       COMN
C           COMPUTES   PAYOFF  IMPROVEMENT                              COMN
C                                                                       COMN
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)          ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN      ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3        XCODES
      COMMON/GLOBAL/                                                    GLOBAL
     *GR         ,ER        ,OMGZ       ,XLAMRF     ,YAURF     ,LUM       GLOBAL
     *,JJOP(10) ,IFATAL    ,NARC         ,NBRAN     ,NFARC     ,ID(4)     GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG     ,MAXTAB                                GLOBAL
     *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)      GLOBAL
     *,ITPSO ,KSOL      ,KGLOBL(8)                                        RETAP
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED    ,IRATED     ,                                           , GENF
     *P1         ,P2        ,P3      ,XK1         ,XK2       ,XK3       , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2D      ,XK3O      , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV         ,PG        ,PP        ,PR        ,PD        ,DPOY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTR ,                 MACH, MACHR,      GENF
     * ISP, TSPF, MACHV,LIFTV , IRATED                                    FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON/STS/                                                       STS
     *OPAY       ,PMIN      ,WORK (20),NWDS       ,IPC    (7),NITER     , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)            STS
      DIMENSION B(9,9), XM(8,8),SQ(8)  ,JJ(9),DSQ(9)                    PAYO2
      COMMON/PARAM/                                                     PARAM
     *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),            PARAM
     *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                       PARAM
     *,DELP(9)                                                          PARAM
      COMMON/SIZING/ DUMSZ(271)                                         AAA
      COMMON/SIZING/ CSIZE(47),SVDPSQ,SVDCON                            PHISZ
      EQUIVALENCE(IPASS,CSIZE(20))                                      PHISZ
      EQUIVALENCE (WORK(5),DQ)                                          PAYO2
C        I    TEST PARAMETERS  IF NONE  GO  TO II                       COMN
C              ELSE  COMPUTE  S MATRIX  CONTRIBUTION TO  A MATRIX       COMN
      IJOP=0                                                            FINI
      IF(NPARA.EQ.0) GO TO 30                                           PAYO2   30
      IF(I2OP.EQ 1 AND.IPASS.LE.2) CALL PARAWT                          PHISZ
      DO 20 I=1,NCN                                                     PAYO2
      DO 20 N=1,NCN                                                     PAYO2
      SS=0.                                                             PAYO2
      DO 10 J=1,NPARA                                                   PAYO2
```

20 OCT 72 S.01-46

```
      SS= SS + SPARA(I,J)*SPARA(N,J)/ WTP(J)                     PAY02
   10 CONTINUE                                                   PAY02
      S2INV(I,N) = SS                                            PAY02
   20 CONTINUE                                                   PAY02
   30 CONTINUE                                                   PAY02
C        II    TEST  FOR  MATRIX  COMPRESSION                    COMN
      NO =0                                                      PAY02
      DO 50 I=1,NCN                                              PAY02
      IF(A(I,I).EQ.0..AND.S2INV(I,I).EQ.0.) GO TO 40             PAY02
      JJ(I)=1                                                    PAY02
      NO =NO +1                                                  PAY02
      DSQ(NO)= DCON(I)                                           PAY02
      GO TO 50                                                   PAY02
   40 JJ(I)=0                                                    PAY02
   50 CONTINUE                                                   PAY02
C        III   STORE  COMPLETE  COMPRESSED  MATRIX IN B          COMN
      IA=0                                                       PAY02
      DO  70  I=1,NCN                                            PAY02
      IF(JJ(I).EQ.0) GO TO 70                                    PAY02
      IA =IA+1                                                   PAY02
      IB = IA-1                                                  PAY02
      DO 60 J=I,NCN                                              PAY02
      IF( JJ(J).EQ.0) GO TO60                                    PAY02
      IB= IB +1                                                  PAY02
      B(IA,IB)= A(I,J) -S2INV(I,J)                               PAY02
   60 CONTINUE                                                   PAY02
   70 CONTINUE                                                   PAY02
C        IV    INVERT  B                                         COMN
      IF(NO GT.1) GO TO 80                                       PAY02
      B(1,1)=1./B(1,1)                                           PAY02
      GO TO 110                                                  PAY02
   80 CONTINUE                                                   PAY02
      CALL SYMVRT( B,NO,IER)                                     PAY02
      IF(IER.NE.0) GO TO 270                                     PAY02
C        V     RESTORE  B   TO  FULL  SIZE                       COMN
      DO 100  I=1,NO                                             PAY02
      DO 90  J=I,NO                                              PAY02
   90 B(I,J) = B(J,I)                                            PAY02
  100 CONTINUE                                                   PAY02
  110 CONTINUE                                                   PAY02
C        VI    IF  NOT  FIRST  OPT  PASS  GO TO  VI-B            COMN
      IF(I2OP.NE.1) GO TO 120                                    PAY02
      IF(IPASS.LE.2) GO TO 115                                   PH1SZ
C          DURING SIZING LOOP RESET METRIC                       PH1SZ
      DPSQ=SVDPSQ                                                PH1SZ
      IF(ABS(DPSQ).LT..005) DPSQ=-.005                           PH1SZ
      DQ=DPAY                                                    PH1SZ
       IJOP=1                                                    FINI
      I2OP =2                                                    PH1SZ
      GO TO 120                                                  PH1SZ
  115 CONTINUE                                                   PH1SZ
C        VI-A   FIRST  OPT. PASS                                 COMN
      DPSQ= DCON(NCN)**2*B(NO,NO)                                PAY02
      I2OP =2                                                    PAY02
      DQ=DCON(NCN)                                               PAY02
      GO TO 190                                                  PAY02
C                                                                COMN
C        VI-B  NORMAL  LOGIC                                     COMN
C                  IF  CONSTRAINTS  GO TO  VI-E                  COMN
  120 CONTINUE                                                   PAY02
      IF(NO.GT.1) GO TO 200                                      PAY02
C        VI-C      PAYOFF ONLY                                   COMN
      DC = SQRT(DPSQ /B(NO,NO)) *SIG                             PAY02
      IF (ABS(DC ) LT.PMIN) ITER =3                              PAY02
      IF ( SIGN(1.,DC ).NE.SIG) ITER =3                          PAY02
  130 IF (ITER.EQ.3) GO TO 140                                   PAY02
C     VI-D     PAYOFF STEP  CONVERGENCE  LOGIC                   COMN
C             COMPARE CURRENT STEP  WITH  LAST STEP , IF GREATER COMN
C             THAN .85  ANL LESS THAN 1.  SCALE UP METRIC AND USE OLD DQ COMN
C             IF GREATER THAN 3.X DQ  LIMIT TO 3.                COMN
```

20 OCT 72 G 01-46

```
C               OTHERWISE USE  CALCULATED VALUE  DC  AND SAVE IN DQ          COMN
C               FOR  NEXT  ITERATION.                                        COMN
          RAT = DC/DQ                                                        PAYO2
          IF(RAT GT  85) GO TO 150                                           UH
      140 DCON(NCN) = DC                                                     PAYO2
          GO TO  180                                                         PAYO2
      150 CONTINUE                                                           PAYO2
          IF(RAT LE.1.) GO TO 160                                            PAYO2
          IF(RAT LE.3 ) GO TO 140                                            UH
          DCON(NCN)=3.*DQ                                                    UH
          GO TO 180                                                          PAYO2
      160 CONTINUE                                                           PAYO2
          DPSQ=DPSQ/(1.85-RAT)**2                                            UH
      170 DCON(NCN) =DQ                                                      PAYO2
      180 CONTINUE                                                           PAYO2
          DQ = DCON(NCN)                                                     PAYO2
      190 CONTINUE                                                           PAYO2
C               PRINT    CALCULATED  STEP  AND  METRIC   AND  RETURN         COMN
          CALL IPR(4HDPSQ,DPSQ,I,1,0)                                        PAYO2
          CALL IPR(2HDC,DC,I,1,0)                                            PAYO2
          RETURN                                                             PAYO2
C         VI-E CALCULATE  PAYOFF  IMPROVE  ACCOUNTING  FOR  CONSTRAINT       COMN
C             MISSES                                                         COMN
      200 DP =0.                                                             PAYO2
          NCC = NO -1                                                        PAYO2
          DO 210  I=1,NCC                                                    PAYO2
      210 DP = DP -B(I,NO)* DSQ(I)                                           PAYO2
          DP = DP / B(NO,NO)                                                 PAYO2
          DO 230 I=1,NCC                                                     PAYO2
          DO 220 J=1,NCC                                                     PAYO2
      220 XM(I,J) = B(I,J) - B(I,NO) * B(J,NO)/ B(NO,NO)                     PAYO2
      230 CONTINUE                                                           PAYO2
          DO 250 I=1,NCC                                                     PAYO2
          SQ(I) =0.                                                          PAYO2
          DO 240  JK= 1,NCC                                                  PAYO2
      240 SQ(I) = SQ(I) + DSQ(JK) * XM(JK,I)                                 PAYO2
      250 CONTINUE                                                           PAYO2
          XM =0.                                                             PAYO2
          DO  260  I=1,NCC                                                   PAYO2
      260 XM = XM + SQ(I) * DSQ(I)                                           PAYO2
          IF(DPSQ-XM.GT.0.) GO TO 290                                        PAYO2
          DC = DP + SIG * SQRT( ( DPSQ -XM)/ B(NO,NO))                       PAYO2
C         VI-F      ON  SIZING PROBLEMS  IJOP  WILL BE NON-ZERO              COMN
C                   ON  SECONDARY PASSES  THROUGH TRAJ. PROGRAM              COMN
C                   SINCE THE OLD  TRAJ. IS  OPTIMIZED  STEP SIZE            COMN
C                   AND  METRIC MUST BE  SMALL  BUT NOT EXCESSIVELY          COMN
C                   THIS  LOGIC ASSURES  SENSIBLE VALUES                     COMN
          IF(IJOP.EQ 0) GO TO 265                                            FINI
          IF(ABS(DC) GT.3.*PMIN) GO TO 265                                   FINI
          DC=3 *PMIN*SIG                                                     FINI
          DPSQ=9.*DPSQ                                                       FINI
          GO TO 130                                                          FINI
C          VII     PROBLEM  CONVERGANCE  TEST                                COMN
      265 CONTINUE                                                           FINI
          IF (ABS (DC ).LT. PMIN) ITER =3                                    PAYO2
C          VII-A    CONSTRAINT  MISSES CAUSING  TROUBLE  TEST                COMN
C                   GO TO  VII-B                                             COMN
          IF(SIGN(1 ,DC).NE.SIG) GO TO 290                                   PAYO2
          GO TO 130                                                          PAYO2
      270 WRITE(6,280)NO,((B(I,J),J=1,9),I=1,9 )                             PAYO2
      280 FORMAT (////,2X,31H MTX INVERS. ERROR IN PAYO NO =I3/,(E15 6,8E13. PAYO2
         16))                                                                PAYO2
          CALL STPIT(11)                                                     PAYO2
C          VII-B      CONSTAINT MISS PROBLEM                                 COMN
C                       TEST  IF  IT HAPPENED  BEFORE  IF SO  QUIT           COMN
C                       ELSE  RESET  TO  RESTORE  CONSTRAINTS  AND           COMN
C                       PRINT  MESSAGE                                       COMN
      290 IF(I2OP NE 3) GO TO 300                                            PAYO2
          ITER =3                                                            PAYO2
          GO TO 140                                                          PAYO2
```



```
      300 IZOP = 3                                                        PAYO2
          ITER=1                                                          PAYO2
          NCN= NCNST                                                      PAYO2
          WRITE(6,310)                                                    PAYO2
      310 FORMAT     (    52HO PAYOFF LOSS PREDICTED, REDUCE CONSTRAINT MISSE PAYO2
         1S                )                                              PAYO2
          RETURN                                                          PAYO2
          END                                                             PAYO2
```

20 OCT 72 6.01-46

SUBRØUTINE PDBC

Subroutine PDBC

Purpose

This subroutine computes functions of state and a partial derivative for cut-off, constraint and payoff targets.

Description

PDBC (and PDBCQL in the quasi-linearization module) contains the equations for all function target conditions available in the trajectory program. These target conditions include the orbital injection parameters, entry range and special targets described in Section 5 of Volume I.

A key feature of the logic in this routine is concerned with the dependance of orbital parameters on one another. This is accommodated by using masking functions to trigger computation of the parameters needed to calculate the desired function.

An example might be when PDBC is called with the variable code argument, KK = 17 meaning eccentricity is to be calculated. The equation sequence necessary for this computation is:

```
ECC  = SQRT (1. - RV20MU * 2. - RV20MU * CSGI * CSGI)
       (RV2ØMU is a function of VI)

CSGJ = SQRT (1 - SNGI * SNGI)

SNGI = V * SINGAM/VI

  VI = SQRT (V * V + 2. * VE * V * COSGAM * SINPSI + VE * VE)
```

Therefore the masking of variable MM and MASK1, MASK2, and MASK6 is non zero and MM is less than MASK7.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AINCL	i	M	Orbital inclination	(RAD)	/ORBIT /(	7)	OUT	I	AINCL
							PDBC	M	AINCL
ALT	h	I	Altitude		/STATE3/(	3)	EQUA3	I	ALT
							OUT	I	ALT
							PDBC	I	ALT
ANOMLY	ζ	M	True anomaly	(RAD)	/ORBIT /(	13)	OUT	I	ANOMLY
							PDBC	M	ANOMLY
APOGEE	R_a	O	Apogee radius	(FT)	/ORBIT /(	11)	OUT	I	APOGEE
							PDBC	O	APOGEE
ARGP	ω_p	M	Orbital argument of perigee	(RAD)	/ORBIT /(	8)	OUT	I	ARGP
							PDBC	M	ARGP
ASCNOD	Ω	M	Longitude of ascending node	(RAD)	/ORBIT /(	9)	OUT	I	ASCNOD
							PDBC	M	ASCNOD
ASYMP	θ	M	Outgoing asymptote	(RAD)	/ORBIT /(	16)	PDBC	M	ASYMP
CAPX	X	O	Asymptote parameter	T	/ORBIT /(	14)	PDBC	O	CAPX
CAPY	Y	O	Asymptote parameter	(FT)	/ORBIT /(	15)	PDBC	O	CAPY
COSOMU	$\cos(\mu-\mu_r)$	M	See symbol		/ORBIT /(	163)	PDBC	M	COSOMU
COSGAM	$\cos(\gamma)$	I	See symbol		/STATE3/(	687)	ACCEL	I	COSGAM
							BL4	I	COSGAM
							BL8	I	COSGAM
							DER3A	I	COSGAM
							EQUA3	O	COSGAM
							MODELA	I	COSGAM
							MODELB	I	COSGAM
							OUT	I	COSGAM
							PDBC	I	COSGAM
							PDY3A	I	COSGAM
COSPSI	$\cos(\psi)$	I	See symbol		/STATE3/(	705)	BL4	I	COSPSI
							BL7	I	COSPSI
							BL8	I	COSPSI
							DER3A	I	COSPSI
							EQUA3	O	COSPSI
							MODELA	I	COSPSI
							MODELB	I	COSPSI
							PDBC	I	COSPSI
							PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol		/STATE3/(	707)	BL4	I	COSRHO
							BL7	I	COSRHO
							BL8	I	COSRHO
							DER3A	I	COSRHO
							EQUA3	M	COSRHO
							MODELA	I	COSRHO
							MODELB	I	COSRHO
							OUT	I	COSRHO
							PDBC	I	COSRHO
							PDY3A	I	COSRHO
CPSII	$\cos(\psi_I)$	M	Cosine of inertial azimuth		/ORBIT /(	157)	PDBC	M	CPSII
CSANO	$\cos(\bar{\zeta})$	M	See symbol		/ORBIT /(	162)	PDBC	M	CSANO
CSGI	$\cos(\gamma_I)$	M	Cosine of inertial flight path angle		/ORBIT /(	155)	PDBC	M	CSGI
CSI	$\cos(i)$	M	Cosine of inclination		/ORBIT /(	159)	PDBC	M	CSI
CSPSR	$\cos(\psi_r)$	I	Cosine of reference azimuth		/ORBIT /(	153)	PDBC	I	CSPSR
							REU3	O	CSPSR
CSXLMR	$\cos(\rho-\rho_r)$	I	Cosine of refernece latitude		/ORBIT /(	147)	PDBC	I	CSXLMR
							REU3	O	CSXLMR
DANDG		O	Partial derivative of boundary condition		/ORBIT /(	104)	PDBC	O	DANDG
DANDH		O	Partial derivative of boundary condition		/ORBIT /(	105)	PDBC	O	DANDH
DANDM		O	Partial derivative of boundary condition		/ORBIT /(	106)	PDBC	O	DANDM
DANDMU		O	Partial derivative of boundary condition		/ORBIT /(	109)	PDBC	O	DANDMU

8 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DANDPS		0	Partial derivative of boundary condition	/ORBIT /(	107)	PDBC	0	DANDPS
DANDRO		0	Partial derivative of boundary condition	/ORBIT /(	108)	PDBC	0	DANDRO
DANDV		0	Partial derivative of boundary condition	/ORBIT /(	103)	PDBC	0	DANDV
DAPDG		0	Partial derivative of boundary condition	/ORBIT /(	90)	PDBC	0	DAPDG
DAPDH		0	Partial derivative of boundary condition	/ORBIT /(	91)	PDBC	0	DAPDH
DAPDM		0	Partial derivative of boundary condition	/ORBIT /(	92)	PDBC	0	DAPDM
DAPDMU		0	Partial derivative of boundary condition	/ORBIT /(	95)	PDBC	0	DAPDMU
DAPDPS		0	Partial derivative of boundary condition	/ORBIT /(	93)	PDBC	0	DAPDPS
DAPDRO		0	Partial derivative of boundary condition	/ORBIT /(	94)	PDBC	0	DAPDRO
DAPDV		0	Partial derivative of boundary condition	/ORBIT /(	89)	PDBC	0	DAPDV
DASDG		0	Partial derivative of boundary condition	/ORBIT /(	125)	PDBC	0	DASDG
DASDH		0	Partial derivative of boundary condition	/ORBIT /(	126)	PDBC	0	DASDH
DASDM		0	Partial derivative of boundary condition	/ORBIT /(	127)	PDBC	0	DASDM
DASDMU		0	Partial derivative of boundary condition	/ORBIT /(	130)	PDBC	0	DASDMU
DASDPS		0	Partial derivative of boundary condition	/ORBIT /(	128)	PDBC	0	DASDPS
DASDRO		0	Partial derivative of boundary condition	/ORBIT /(	129)	PDBC	0	DASDRO
DASDV		0	Partial derivative of boundary condition	/ORBIT /(	124)	PDBC	0	DASDV
DBEDG		0	Partial derivative of boundary condition	/ORBIT /(	69)	PDBC	0	DBEDG
DBEDH		0	Partial derivative of boundary condition	/ORBIT /(	70)	PDBC	0	DBEDH
DBEDM		0	Partial derivative of boundary condition	/ORBIT /(	71)	PDBC	0	DBEDM
DBEDMU		0	Partial derivative of boundary condition	/ORBIT /(	74)	PDBC	0	DBEDMU
DBEDPS		0	Partial derivative of boundary condition	/ORBIT /(	72)	PDBC	0	DBEDPS
DBEDRO		0	Partial derivative of boundary condition	/ORBIT /(	73)	PDBC	0	DBEDRO
DBEDV		0	Partial derivative of boundary condition	/ORBIT /(	68)	PDBC	0	DBEDV
DCXDG		0	Partial derivative of boundary condition	/ORBIT /(	111)	PDBC	0	DCXDG
DCXDH		0	Partial derivative of boundary condition	/ORBIT /(	112)	PDBC	0	DCXDH
DCXDM		0	Partial derivative of boundary condition	/ORBIT /(	113)	PDBC	0	DCXDM
DCXDMU		0	Partial derivative of boundary condition	/ORBIT /(	116)	PDBC	0	DCXDMU
DCXDPS		0	Partial derivative of boundary condition	/ORBIT /(	114)	PDBC	0	DCXDPS
DCXDRO		0	Partial derivative of boundary condition	/ORBIT /(	115)	PDBC	0	DCXDRO
DCXDV		0	Partial derivative of boundary condition	/ORBIT /(	110)	PDBC	0	DCXDV
DCYDG		0	Partial derivative of boundary condition	/ORBIT /(	118)	PDBC	0	DCYDG
DCYDH		0	Partial derivative of boundary condition	/ORBIT /(	119)	PDBC	0	DCYDH
DCYDM		0	Partial derivative of boundary condition	/ORBIT /(	120)	PDBC	0	DCYDM
DCYDMU		0	Partial derivative of boundary condition	/ORBIT /(	123)	PDBC	0	DCYDMU
DCYDPS		0	Partial derivative of boundary condition	/ORBIT /(	121)	PDBC	0	DCYDPS
DCYDRO		0	Partial derivative of boundary condition	/ORBIT /(	122)	PDBC	0	DCYDRO
DCYDV		0	Partial derivative of boundary condition	/ORBIT /(	117)	PDBC	0	DCYDV
DECDG		M	Partial derivative of boundary condition	/ORBIT /(	55)	PDBC	M	DECDG
DECDH		M	Partial derivative of boundary condition	/ORBIT /(	56)	PDBC	M	DECDH
DECDM		0	Partial derivative of boundary condition	/ORBIT /(	57)	PDBC	0	DECDM
DECDMU		0	Partial derivative of boundary condition	/ORBIT /(	60)	PDBC	0	DECDMU
DECDPS		M	Partial derivative of boundary condition	/ORBIT /(	58)	PDBC	M	DECDPS
DECDRO		M	Partial derivative of boundary condition	/ORBIT /(	59)	PDBC	M	DECDRO
DECDV		M	Partial derivative of boundary condition	/ORBIT /(	54)	PDBC	M	DECDV
DENDG		0	Partial derivative of boundary condition	/ORBIT /(	132)	PDBC	0	DENDG
DENDH		0	Partial derivative of boundary condition	/ORBIT /(	133)	PDBC	0	DENDH
DENDM		0	Partial derivative of boundary condition	/ORBIT /(	134)	PDBC	0	DENDM
DENDMU		0	Partial derivative of boundary condition	/ORBIT /(	137)	PDBC	0	DENDMU

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DENDPS		0	Partial derivative of boundary condition	/ORBIT /(	135)	PDBC	0	DENDPS
DENDRO		0	Partial derivative of boundary condition	/ORBIT /(	136)	PDBC	0	DENDRO
DENDV		0	Partial derivative of boundary condition	/ORBIT /(	131)	PDBC	0	DENDV
DGIDG		M	Partial derivative of boundary condition	/ORBIT /(	27)	PDBC	M	DGIDG
DGIDH		M	Partial derivative of boundary condition	/ORBIT /(	28)	PDBC	M	DGIDH
DGIDM		0	Partial derivative of boundary condition	/ORBIT /(	29)	PDBC	0	DGIDM
DGIDMU		0	Partial derivative of boundary condition	/ORBIT /(	32)	PDBC	0	DGIDMU
DGIDPS		M	Partial derivative of boundary condition	/ORBIT /(	30)	PDBC	M	DGIDPS
DGIDRO		M	Partial derivative of boundary condition	/ORBIT /(	31)	PDBC	M	DGIDRO
DGIDV		M	Partial derivative of boundary condition	/ORBIT /(	26)	PDBC	M	DGIDV
DIDG		M	Partial derivative of boundary condition	/ORBIT /(	62)	PDBC	M	DIDG
DIDH		M	Partial derivative of boundary condition	/ORBIT /(	63)	PDBC	M	DIDH
DIDM		0	Partial derivative of boundary condition	/ORBIT /(	64)	PDBC	0	DIDM
DIDMU		0	Partial derivative of boundary condition	/ORBIT /(	67)	PDBC	0	DIDMU
DIDPS		M	Partial derivative of boundary condition	/ORBIT /(	65)	PDBC	M	DIDPS
DIDRO		M	Partial derivative of boundary condition	/ORBIT /(	66)	PDBC	M	DIDRO
DIDV		M	Partial derivative of boundary condition	/ORBIT /(	61)	PDBC	M	DIDV
DMIDG		0	Partial derivative of boundary condition	/ORBIT /(	41)	PDBC	0	DMIDG
DMIDH		0	Partial derivative of boundary condition	/ORBIT /(	42)	PDBC	0	DMIDH
DMIDM		0	Partial derivative of boundary condition	/ORBIT /(	43)	PDBC	0	DMIDM
DMIDMU		0	Partial derivative of boundary condition	/ORBIT /(	46)	PDBC	0	DMIDMU
DMIDPS		0	Partial derivative of boundary condition	/ORBIT /(	44)	PDBC	0	DMIDPS
DMIDRO		0	Partial derivative of boundary condition	/ORBIT /(	45)	PDBC	0	DMIDRO
DMIDV		0	Partial derivative of boundary condition	/ORBIT /(	40)	PDBC	0	DMIDV
DMODG		0	Partial derivative of boundary condition	/ORBIT /(	139)	PDBC	0	DMODG
DMODH		0	Partial derivative of boundary condition	/ORBIT /(	140)	PDBC	0	DMODH
DMODM		0	Partial derivative of boundary condition	/ORBIT /(	141)	PDBC	0	DMODM
DMODMU		0	Partial derivative of boundary condition	/ORBIT /(	144)	PDBC	0	DMODMU
DMODPS		0	Partial derivative of boundary condition	/ORBIT /(	142)	PDBC	0	DMODPS
DMODRO		0	Partial derivative of boundary condition	/ORBIT /(	143)	PDBC	0	DMODRO
DMODV		0	Partial derivative of boundary condition	/ORBIT /(	138)	PDBC	0	DMODV
DNODG		0	Partial derivative of boundary condition	/ORBIT /(	76)	PDBC	0	DNODG
DNODH		0	Partial derivative of boundary condition	/ORBIT /(	77)	PDBC	0	DNODH
DNODM		0	Partial derivative of boundary condition	/ORBIT /(	78)	PDBC	0	DNODM
DNODMU		0	Partial derivative of boundary condition	/ORBIT /(	81)	PDBC	0	DNODMU
DNODPS		0	Partial derivative of boundary condition	/ORBIT /(	79)	PDBC	0	DNODPS
DNODRO		0	Partial derivative of boundary condition	/ORBIT /(	80)	PDBC	0	DNODRO
DNODV		0	Partial derivative of boundary condition	/ORBIT /(	75)	PDBC	0	DNODV
DPDG		M	Partial derivative of boundary condition	/ORBIT /(	48)	PDBC	M	DPDG
DPDH		M	Partial derivative of boundary condition	/ORBIT /(	49)	PDBC	M	DPDH
DPDM		0	Partial derivative of boundary condition	/ORBIT /(	50)	PDBC	0	DPDM
DPDMU		0	Partial derivative of boundary condition	/ORBIT /(	53)	PDBC	0	DPDMU
DPDPS		M	Partial derivative of boundary condition	/ORBIT /(	51)	PDBC	M	DPDPS
DPDRO		M	Partial derivative of boundary condition	/ORBIT /(	52)	PDBC	M	DPDRO
DPDV		M	Partial derivative of boundary condition	/ORBIT /(	47)	PDBC	M	DPDV
DPEDG		0	Partial derivative of boundary condition	/ORBIT /(	97)	PDBC	0	DPEDG
DPEDH		0	Partial derivative of boundary condition	/ORBIT /(	98)	PDBC	0	DPEDH
DPEDM		0	Partial derivative of boundary condition	/ORBIT /(	99)	PDBC	0	DPEDM
DPEDMU		0	Partial derivative of boundary condition	/ORBIT /(	102)	PDBC	0	DPEDMU

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DPEDPS		O	Partial derivative of boundary condition	/ORBIT /(	100)	PDBC	O	DPEDPS
DPEDRO		O	Partial derivative of boundary condition	/ORBIT /(	101)	PDBC	O	DPEDRO
DPEDV		O	Partial derivative of boundary condition	/ORBIT /(	96)	PDBC	O	DPEDV
DPIDG		M	Partial derivative of boundary condition	/ORBIT /(	34)	PDBC	M	DPIDG
DPIDH		M	Partial derivative of boundary condition	/ORBIT /(	35)	PDBC	M	DPIDH
DPIDM		O	Partial derivative of boundary condition	/ORBIT /(	36)	PDBC	O	DPIDM
DPIDMU		O	Partial derivative of boundary condition	/ORBIT /(	39)	PDBC	O	DPIDMU
DPIDPS		M	Partial derivative of boundary condition	/ORBIT /(	37)	PDBC	M	DPIDPS
DPIDRO		M	Partial derivative of boundary condition	/ORBIT /(	38)	PDBC	M	DPIDRO
DPIDV		M	Partial derivative of boundary condition	/ORBIT /(	33)	PDBC	M	DPIDV
DSMDG		M	Partial derivative of boundary condition	/ORBIT /(	83)	PDBC	M	DSMDG
DSMDH		M	Partial derivative of boundary condition	/ORBIT /(	84)	PDBC	M	DSMDH
DSMDM		O	Partial derivative of boundary condition	/ORBIT /(	85)	PDBC	O	DSMDM
DSMDMU		O	Partial derivative of boundary condition	/ORBIT /(	88)	PDBC	O	DSMDMU
DSMDPS		M	Partial derivative of boundary condition	/ORBIT /(	86)	PDBC	M	DSMDPS
DSMDRO		M	Partial derivative of boundary condition	/ORBIT /(	87)	PDBC	M	DSMDRO
DSMDV		M	Partial derivative of boundary condition	/ORBIT /(	82)	PDBC	M	DSMDV
DVIDG		M	Partial derivative of boundary condition	/ORBIT /(	20)	PDBC	M	DVIDG
DVIDH		M	Partial derivative of boundary condition	/ORBIT /(	21)	PDBC	M	DVIDH
DVIDM		O	Partial derivative of boundary condition	/ORBIT /(	22)	PDBC	O	DVIDM
DVIDMU		O	Partial derivative of boundary condition	/ORBIT /(	25)	PDBC	O	DVIDMU
DVIDPS		M	Partial derivative of boundary condition	/ORBIT /(	23)	PDBC	M	DVIDPS
DVIDRO		M	Partial derivative of boundary condition	/ORBIT /(	24)	PDBC	M	DVIDRO
DVIDV		M	Partial derivative of boundary condition	/ORBIT /(	19)	PDBC PDBC	M I	DVIDV PPD
ECC	e	M	Orbital eccentricity	/ORBIT /(	6)	OUT PDBC	I M	ECC ECC
ENERGY	E	O	Energy	/ORBIT /(	17)	OUT PDBC	I O	ENERGY ENERGY
ER	E_R	I	Earth radius. (FT)	/GLOBAL/(	2)	COORDS CRASH EQUA3 GEINP PADS1 PDBC SOMG TRTOSZ	I I I I I I I I	ER REM ER ER ER ER ER ER
F		M	Value of boundary condition function or array of state derivatives	/PDBC /(*	)	PDBC	M	F
GAMI	γ_I	O	Inertial flight path angles (RAD)	/ORBIT /(	2)	OUT PDBC	I O	GAMI GAMI
GD	$\dot{\gamma}$	I	GAM derivative	/STATE3/(	16)	DER3A PDBC	O I	GD GD
GM	GM	I	Product of Newton's universal gravitational constant and the mass of the earth. (FT^3/SEC^2)	/GLOBAL/(	67)	CRASH OUT PADS1 PDBC	I I O I	GM GM GM GM

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
HD	$\dot{h}$	I	ALT derivative	/STATE3/(	17)	DER3A	O	HD
						PDBC	I	HD
HMNTM	H	M	Momentum	/ORBIT /(	18)	OUT	I	HMNTM
						PDBC	M	HMNTM
HTD	$\dot{Q}$	I	Heating derivative	/STATE3/(	22)	DER3A	O	HTD
						OUT	I	HTD
						PDBC	I	HTD
						PDY3A	M	HTD
IATM		I	Atmosphere option flag	/ARCDAT/(	7)	EQUA3	I	IATM
						FXDAT	I	IATM
						OUT	I	IATM
						PDBC	I	IATM
						VT	I	IATM
IOP		I	Option code	/PDBC /(*	)	PDBC	I	IOP
IPFLG1		I	IPFLG1≠0 supresses print-out of velocity losses and inertial Euler angles.	/GLOBAL/(	69)	FNTG	I	IPFLG1
						OUT	I	IPFLG1
						PDBC	I	IPFLG1
						PRINT	I	IPFLG1
						TRTOSZ	O	IPFLG1
ISKP		M	Redundancy flag	/PDBC /(*	)	PDBC	M	ISKP
JVAR		C	Masking array counterpart (powers of 2)	/PDBC /(*	)	PDBC	C	JVAR
K		W	Variable code number	/PDBC /(*	)	PDBC	W	K
KK		I	Variable code number	/PDBC /(*	)	PDBC	I	KK
MASK		D	Masking array for dependency calculation of orbit parameters and partial derivatives	/PDBC /(*	)	PDBC	D	MASK
MU	μ	I	Longitude	/STATE3/(	7)	OUT	I	MU
						PDBC	I	MU
OCORHO	$\omega\times$COSRHO	I	See symbol	/STATE3/(	708)	DER3A	I	OCORHO
						EQUA3	M	OCORHO
						MODELA	I	OCORHO
						MODELB	I	OCORHO
						PDBC	I	OCORHO
						PDY3A	I	OCORHO
OMGZ	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
						CRASH	I	OMEGA
						DER3A	I	OMGZ
						EQUA3	I	OMGZ
						GEINP	I	OMGZ
						MODELA	I	OMGZ
						MODELB	I	OMGZ
						PDBC	I	OMGZ
						PDY3A	I	OMGZ
						SDINP	I	OMGZ
						TOPM	I	OMGZ
ORBPRM	V_I	I	Inertial velocity (FT/SEC)	/ORBIT /(	1)	OUT	I	VI
						PDBC	I	ORBPRM
						PDBC	M	VI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
P	p_r	M	Semi-latus rectum	(FT)	/ORBIT /(	5)	OUT	I	P
							PDBC	M	P
PA	p_a	I	Atmospheric pressure	(PSF)	/GENF /(	308)	EQUA3	M	DZM
							FH2	I	PA
							IMPUL	I	PA
							OUT	I	PA
							PDBC	I	PA
							SDER3	I	PA
PERGEE	R_p	O	Perigee radius	(FT)	/ORBIT /(	12)	OUT	I	PERGEE
							PDBC	O	PERGEE
PPD		I	Partial derivative of boundary condition		/ORBIT /(	19)	PDBC	M	DVIDV
							PDBC	I	PPD
PSII	ψ_I	M	Inertial azimuth	(RAD)	/ORBIT /(	3)	OUT	I	PSII
							PDBC	M	PSII
Q	q	I	Dynamic pressure	(PSF)	/GENF /(	303)	ENVPRM	I	Q
							EQUA3	M	Q
							OUT	I	Q
							PDBC	I	Q
							VT	I	Q
R	R	I	Radial distance from earth center to vehicle	(FT)	/GENF /(	305)	BL4	I	R
							BL7	I	R
							BL8	I	R
							DER3A	I	R
							EQUA3	M	R
							MODELA	I	R
							MODELB	I	R
							PDBC	I	R
							PDY3A	I	R
							TRTOSZ	I	R
RE	R_{ey}	M	Unit reynolds number	(1/FT)	/GENF /(	306)	OUT	M	RE
							PDBC	M	RE
RO	ρ_a	I	Atmospheric density	(SLUGS/FT**3)	/GENF /(	309)	BL7	I	RO
							BL8	I	RO
							DER3A	I	RO
							EQUA3	I	RO
							OUT	I	RO
							PDBC	I	RO
							PDY3A	I	RO
ROR		I	Deriv. Of density wrt alt.		/GENF /(	313)	BL7	I	ROR
							BL8	I	ROR
							EQUA3	I	ROR
							PDBC	I	ROR
							PDY3A	I	ROR
S		M	Partials of boundary conditions		/PDBC /(*	)	PDBC	M	S
SCROSS	S_c	O	Cross range	(FT)	/ORBIT /(	149)	OUT	I	SCROSS
							PDBC	O	SCROSS
SD		M	Rate of change of boundary condition function		/PDBC /(*	)	PDBC	M	SD
SDOWN	S_D	O	Down range	(FT)	/ORBIT /(	148)	OUT	I	SDOWN
							PDBC	O	SDOWN
SINDMU	$\sin(\mu-\mu_r)$	M	See symbol		/ORBIT /(	164)	PDBC	M	SINDMU
SINGAM	$\sin(\gamma)$	I	See symbol		/STATE3/(	688)	BL4	I	SINGAM
							BL7	I	SINGAM
							BL8	I	SINGAM
							DER3A	I	SINGAM
							EQUA3	O	SINGAM
							MODELA	I	SINGAM
							MODELB	I	SINGAM
							PDBC	I	SINGAM
							PDY3A	I	SINGAM
							SDER3	I	SINGAM

8 NOV 72 6 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
SINPSI	$\sin(\psi)$	I	See symbol	/STATE3/(	704)	BL4	I	SINPSI
						BL7	I	SINPSI
						BL8	I	SINPSI
						DER3A	I	SINPSI
						EQUA3	O	SINPSI
						MODELA	I	SINPSI
						MODELB	I	SINPSI
						PDBC	I	SINPSI
						PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol	/STATE3/(	706)	BL4	I	SINRHO
						BL7	I	SINRHO
						BL8	I	SINRHO
						DER3A	I	SINRHO
						EQUA3	O	SINRHO
						MODELA	I	SINRHO
						MODELB	I	SINRHO
						OUT	I	SINRHO
						PDBC	I	SINRHO
						PDY3A	I	SINRHO
SMIMAJ	a_s	M	Semi-major axis (FT)	/ORBIT /(	10)	OUT	I	SMIMAJ
						PDBC	M	SMIMAJ
SNGI	$\sin(\gamma_I)$	M	Sine of inertial flight path angle	/ORBIT /(	154)	PDBC	M	SNGI
SNGNU	$\sin(\nu)$	M	See symbol	/ORBIT /(	161)	PDBC	M	SNGNU
SNI	$\sin(i)$	M	Sine of inclination	/ORBIT /(	160)	PDBC	M	SNI
SNPSR	$\sin(\psi_r)$	I	Sine of reference azimuth	/ORBIT /(	152)	PDBC	I	SNPSR
						REU3	O	SNPSR
SNXLMR	$\sin(\rho-\rho_r)$	I	Sine of reference latitude	/ORBIT /(	146)	PDBC	I	SNXLMR
						REU3	O	SNXLMR
SPSII	$\sin(\psi_I)$	M	Sin of inertial azimuth	/ORBIT /(	156)	PDBC	M	SPSII
SQ		I	A synthesis data array (37,5) that contains the flyback data and some injection quantities	/SIZING/(	74)	ENVPRM	M	SQ
						FLYBKP	M	SQ
						ISPRAT	I	SQ
						PDBC	I	SQ
						PRITVA	I	SQ
						RANGE	M	SQ
						REU3	O	SQ
						SIZE	O	SQ
						SIZEMR	M	SQ
						SIZIN	M	SQ
						STAU	I	SQ
						SUMOUT	M	SQ
						TAMPAR	O	SQ
						TAMPER	M	SQ
						THRUST	M	SQ
						TRTOSZ	M	SQ
						VEHDF	M	SQ
						WTVOL	M	SQ
SQRT		F	Square root function	/SQRT /($	)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTOE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAY02	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
STOT	S_T	O	Total range (FT)	/ORBIT /(	158)	OUT	I	STOT
						PDBC	O	STOT
						TRTOSZ	I	STOT

8 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
TC	θ_C	M	Cross range angle	(RAD)	/ORBIT /(	151)	PDBC	M	TC
TD	θ_D	M	Down range angle	(RAD)	/ORBIT /(	150)	PDBC	M	TD
THT	θ_T	M	Total range angle	(FT)	/ORBIT /(	165)	PDBC	M	THT
TIME	t	I	Time (elapsed)		/GENF /(	493)	ADICB3	O	TIME
							AST3	I	TIME
							BNTG	M	TIME
							CON3	I	TIME
							DTF3	I	TIME
							ENVPRM	I	TIME
							EQUA3	I	TIME
							FNTG	M	TIME
							MODELA	I	TIME
							OUT	I	TIME
							PDBC	I	TIME
							PROPIN	I	TIME
							REU3	M	TIME
							RKTA3A	M	TT
							RKTB3A	M	TT
							YREF3	M	TIME
UD	$\dot{\mu}$	I	Longitude derivative		/STATE3/(	21)	DER3A	O	UD
							PDBC	I	UD
V	v	I	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	Y
							STP3	I	VAR
							TOPM	O	KWOW
							YREF3	M	V
VD	$\dot{y}$	I	State vector derivatives in steepest descent module		/STATE3/(	15)	ADICB3	M	DVAR
							ADIC3A	I	DVAR
							ADID3A	M	DVAR
							DER3A	O	VD
							DTF3	I	VT
							ENVPRM	I	DVAR
							PDBC	I	VD
							PROPIN	O	DVAR
							REU3	I	DVAR
							RKTA3A	I	DY
							SDER3	O	DVAR
							STP3	I	DVAR
							YREF3	I	DVAR
							YREF3	I	VT
VE	$R\omega\cos\rho$	W	See symbol		/PDBC /(*	)	PDBC	W	VE
VI	V_I	M	Inertial velocity	(FT/SEC)	/ORBIT /(	1)	OUT	I	VI
							PDBC	I	ORBPRM
							PDBC	M	VI
VNR		I	Deriv.of viscosity wrt alt.		/GENF /(	315)	PDBC	I	VNR

8 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
VNU	μ_{a}	I	Atmospheric viscosity [dynamic]	(SLUGS/FT/SEC)	/GENF	/(311)	OUT	I	VNU
							PDBC	I	VNU
W	W	I	Weight	(LBS)	/GENF	/(412)	BL5	I	W
							ENVPRM	I	W
							EQUA3	R	W
							FH3	I	W
							OUT	I	W
							PDBC	I	W
							REU3	I	W
							TRTOSZ	I	W
XMUI	μ_{I}	M	Inertial longitude	(RAD)	/ORBIT	/(4)	OUT	I	XMUI
							PDBC	M	XMUI
YMXRF	ρ_{r}	I	Reference longitude	(RAD)	/ORBIT	/(145)	PDBC	I	YMXRF
							REU3	O	YMXRF

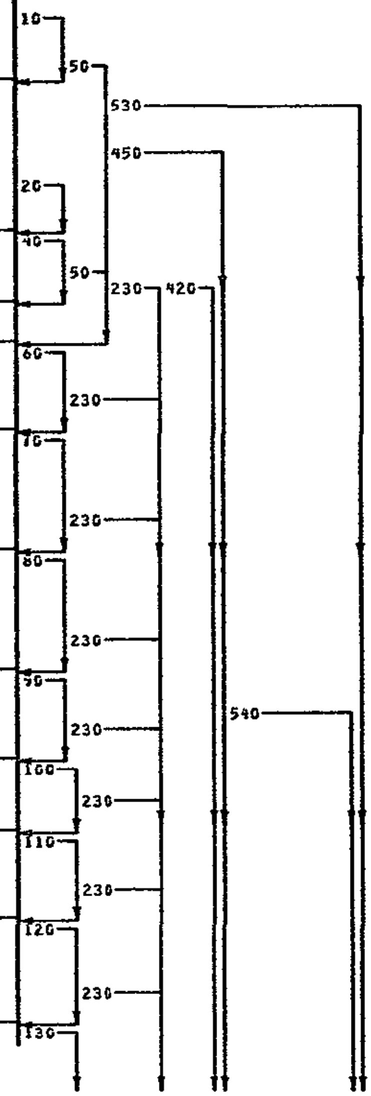
```
      SUBROUTINE PDBC(KK,F,S,SD,IOP,ISKP)                               PDBC
C                                                                       PDBC
C         THIS  SUBROUTINE  COMPUTES  FUNCTIONS OF STATE  AND PARTIALS  PDBC
C         FOR  VARIOUS  PURPOSES  IN  TABTOP                            PDBC
C                 ** DEFINITIONS**                                      PDBC
C                                                                       PDBC
C         KK  =  VARIABLE  CODE  NUMBER                                 PDBC
C         F   =  VALUE OF FUNCTION                                      PDBC
C         S   =  PARTIAL DERIVATIVE  ARRAY                              PDBC
C         SD  =  TIME DERIVATIVE OF F                                   PDBC
C         IOP =  OPTION FLAG = 0 =  COMPUTE F  ONLY                     PDBC
C                              1 =  COMPUTE S                           PDBC
C                              2 =  ESTIMATE SD  IF POSSIBLE            PDBC
C                              3 =  COMPUTE  SD                         PDBC
C         ISKP=  BYPASS FLAG  TO  INHIBIT  REDUNDANT  COMPUTATION       PDBC
C                                                                       PDBC
      DIMENSION S(1),F(1)                                               PDBC
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI      , ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX      , ARCDAT
     *XLMAX     ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA      , ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG      , ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB      , ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT        , ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                  ARCDAT
     * ,FRATE  , ARCD(9)                                                 RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)   , STATE3D
     *XL(9,9)   ,YOP(20,9),YOS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI    ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    , STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                  STATE3D
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP       ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO       ,UDO       ,HTDV      ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                       STATE3D
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,  EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2), EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD), EQUV3
     *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)            EQUV3
      REAL M,MU,MD                                                      EQUV3
      COMMON/GENF/                                                      GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R         ,RE        ,MACH:     ,PA        ,RO        ,CS        , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                  GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED    ,IRATED    ,                                             GENF
     *P1        ,P2        ,P3        ,XK1       ,XK2       ,XK3       , GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)   GENF
```

20 OCT 72 G 01-46

```
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,        GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                    FRAT
      DIMENSION TPH1(10),TST1(10)                                         GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                  GENF
      COMMON/GLOBAL/                                                      GLOBAL
     *GR       ,ER       ,OMGZ     ,XLAMRF     ,YMURF     ,LUM            GLOBAL
     *,JJOP(10) ,IFATAL   ,NARC     ,NBRAN     ,NFARC     ,ID(4)          GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                                 GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)       GLOBAL
     *,ITPSO ,KSOL     ,KGLOBL(8)                                         RETAP
      COMMON/ ORBIT/  VI,      GAMI,     PSII,     XMUI,        P,        ORBIT
     *     ECC,    AINCL,     ARGP,   ASCNOD,   SMIMAJ,   APOGEE,         ORBIT
     *  PERGEE,   ANOMLY,     CAPX,     CAPY,    ASYMP,   ENERGY,         ORBIT
     *   HMNTM,    DVIDV,    DVIDG,                                       ORBIT
     *  DVIDH ,    DVIDM,   DVIDPS,   DVIDRO,   DVIDMU,    DGIDV,         ORBIT
     *  DGIDG ,    DGIDH,   DGIDM ,   DGIDPS,   DGIDRO,    DGIDMU,        ORBIT
     *  DPIDV ,    DPIDG,   DPIDH ,   DPIDM ,   DPIDPS,    DPIDRO,        ORBIT
     *  DPIDMU,    DMIDV,   DMIDG ,   DMIDH ,   DMIDM ,    DMIDPS,        ORBIT
     *  DMIDRO,    DMIDMU,  DPDV  ,   DPDG  ,   DPDH  ,    DPDM  ,        ORBIT
     *  DPDPS ,    DPDRO ,  DPDMU ,   DECDV ,   DECDG ,    DECDH ,     -  ORBIT
     *  DECDM ,    DECDPS,  DECDRO,   DECDMU,   DIDV  ,    DIDG  ,        ORBIT
     *  DIDH  ,    DIDM  ,  DIDPS  ,  DIDRO  ,  DIDMU  ,   DBEDV ,        ORBIT
     *  DBEDG ,    DBEDH ,  DBEDM ,   DBEDPS,   DBEDRO,    DBEDMU,        ORBIT
     *  DNODV ,    DNODG ,  DNODH ,   DNODM ,   DNODPS,    DNODRO,        ORBIT
     *  DNODMU,    DSMDV,    DSMDG,    DSMDH,    DSMDM,   DSMDPS         -ORBIT
      COMMON/ORBIT/                                                       ORBIT
     *  DSMDRO,   DSMDMU,    DAPDV,    DAPDG,    DAPDH,   DAPDM ,         ORBIT-
     *  DAPDPS,   DAPDRO,    DAPDMU,   DPEDV,    DPEDG,   DPEDH ,         ORBIT
     *DPEDM     ,DPEDPS     ,DPEDRO    ,DPEOMU    ,DANDV     ,DANDG     , ORBIT
     *  DANDH ,   DANDM ,    DANDPS,   DANDRO,   DANDMU,  DCXDV ,         ORBIT
     *  DCXDG ,   DCXDH ,    DCXDM ,   DCXDPS,   DCXDRO,  DCXDMU,         ORBIT
     *  DCYDV ,   DCYDG ,    DCYDH ,   DCYDM ,   DCYDPS,  DCYDRO,         ORBIT
     *  DCYDMU,   DASDV ,    DASDG ,   DASDH ,   DASDM -, DASDPS,         ORBIT
     *  DASDRO,   DASDMU,    DENDV ,   DENDG ,   DENDH ,  DENDM ,         ORBIT
     *  DENDPS,   DENDRO,    DENDMU,   DMODV ,   DMODG ,  DMODH ,         ORBIT
     *  DMODM ,   DMODPS,    DMODRO,   DMODMU                             ORBIT
      DIMENSION ORBPRM(18),PPO(7,18)                                      ORBIT
      EQUIVALENCE (VI,ORBPRM),(DVIDV,PPO)                                 ORBIT
      COMMON/ORBIT/ YMXRF,SNXLMR,CSXLMR,SDOWN,SCROSS,TD,TC                ORBIT
     *,SNPSR     ,CSPSR      ,SNGI       ,CSGI       ,SPSII      ,CPSII   , ORBIT
     * STOT      ,CSI        ,SNI        ,SNGNU      ,CSANO      ,COSDMU  , ORBIT
     * SINOMU    ,THT        ,WTFUEL                                      ORBIT
      DIMENSION JVAR(36),MASK(36), MS(2)                                  PDBC
      EQUIVALENCE   (MASK(1),MSK1),(MASK(2),MSK2),(MASK(3),MSK3),         PDBC
     1(MASK(4),MSK4), (MASK(5),MSK5)  ,(MASK(6),MSK6) ,(MASK(7),MSK7),    PDBC
     2(MASK(8),MSK8), (MASK(9),MSK9)  ,(MASK(10),MSK10),(MASK(11),MSK11) PDBC
     3,(MASK(12),MSK12),(MASK(13),MSK13),(MASK(14),MSK14),                PDBC
     4 (MASK(15),MSK15),(MASK(16),MSK16),(MASK(17),MSK17),                PDBC
     5 (MASK(18),MSK18),(MASK(19),MSK19),(MASK(20),MSK20),                PDBC
     6 (MASK(21),MSK21),(MASK(22),MSK22),(MASK(23),MSK23),                PDBC
     7 (MASK(24),MSK24),(MASK(25),MSK25),(MASK(26),MSK26),                PDBC
     8 (MASK(27),MSK27),(MASK(28),MSK28),(MASK(29),MSK29),                PDBC
     9 (MASK(30),MSK30),(MASK(31),MSK31),(MASK(32),MSK32),                PDBC
     X (MASK(33),MSK33),(MASK(34),MSK34),(MASK(35),MSK35),                PDBC
     A (MASK(36),MSK36)                                                   PDBC
      DATA JVAR /0000001,0000003, 0000007,                                PDBC
     1 0000010,0000023, 0000043,                                          PDBC
     2 0000107,   0000367, 0000517, 0001001,                              PDBC
     3 0003043,  0005043,  0010023, 0020063,                              PDBC
     4 0040063, 0100043, 0201001, 0400003,                                PDBC
     501000001     ,03000003      ,04000007      ,010000000     ,         PDBC
     6023000023    ,043000043     ,0104000107    ,0367000367    ,         PDBC
     70504000517   ,01001001001   ,03043001043   ,05043001043   ,         PDBC
     8010023010023 ,020036000063  ,040036000063  ,010004310043  ,         PDBC
     9  01001001001,  01003400003  /                                      PDBC
      DATA  MASK / 1,2, 4, 8, 16, 32 , 64, 128, 256 , 512 , 1024, 2048,   PDBC
     14096, 8192,16384, 32768 , 65536  , 131072 , 262144 ,   524288,      PDBC
     2  1048576 ,    2097152 ,   4194304 , 8388608 , 16777216         ,   PDBC
     3    33554432 , 67108864,  134217728, 268435456,     536870912  ,    PDBC
     4 1073741824 ,  2147483648 ,4294967296,8589934592 ,                  PDBC
     5 17179869184,0/                                                     PDBC
      DATA MS/ 03000003, 07000007/                                        PDBC
      REAL MUB, MUO, ISPB, ISPO, IDVEL,WNB,WO                             SIZING
      COMMON /SIZING/                                                     SIZING
C           PHASE II SIZING PARAMERERS                                    SIZING
```

20 OCT 72 G 01-46

```
      *TZ,      VV(3),      QP(14),   EROR,     PZ(5),     VQ,       SW(20), SIZING
      *SV(28),  SQ(37,5),   SE(11),   TLAT,     TLNG,                       SIZING
C         PHASE I SIZING PARAMERERS                                          SIZING
      *WBO,      WLOO,      DWEB,     DWED,     TOLWT,     WPB,      TWRAT2, SIZING
      *BK1,      BK2,       BK3,      BK4,      ISIZE,     TRAFLG,   TWRATO, SIZING
      *OK1,      OK2,       OK3,      OK4,      PRFLG,     IPASS,    IPSMAX, SIZING
      *AEXIT,    TVACO,     NO,       WFO,      IDVEL,     ISPO,     ISPB,   SIZING
      *XPL,      TVACB,    NNB,       WEO,      WEB,       WO,       WLO,    SIZING
      *DVO,       DVB,       MUB,      MUD,       VSTG,     WPO               SIZING
      *,JTYP,     BECO ,    BSTG ,    ORBI,     ITNBW , ITNOW  ,              SIZING
      * SVDPSQ  , SVDCON  ,IHUNT   ,IOPSTG  ,ISZD(19)                        UH
       IF(KK.GT.0) GO TO  10                                                 PDBC
       K=1                                                                   PDBC
       MM = 0617777                                                          PDBC
       GO TO 50                                                              PDBC
    10 CONTINUE                                                              PDBC
       IF(KK GE 30) GO TO 530                                                PDBC
C           PARAMETER IS  AMONG ORBITAL SET                                  PDBC
C       TEST  FOR  ESTIMATE  OPTION                                          PDBC
       IF(IOP EQ.2) GO TO 450                                                PDBC
       K=KK-11                                                               PDBC
       IF(IOP EQ.0) GO TO 20                                                 PDBC
C              TEST  FOR  PARAMETER CALCULATION ONLY                         PDBC
       K=18+K                                                                PDBC
    20 IF(ISKP.EQ.0) GO TO 40                                                PDBC
    30 MM=AND(JVAR(K),-MS(ISKP))                                             PDBC
       IF(MM.NE.0)  GO TO 50                                                 PDBC
       IF(K-18)230,230,420                                                   PDBC
    40 MM = JVAR(K)                                                          PDBC
C         TEST  AND  COMPUTE  VI                                             PDBC
    50 IF(0.EQ AND(MM,MSK1))  GO  TO  60                                     PDBC
       VE = R* OCORHO                                                        PDBC
       VI=  SQRT( V*V +2.*VE*V*COSGAM*SINPSI+VE*VE)                          PDBC
       IF(MM LT MSK2) GO TO 230                                              PDBC
C         TEST  AND   COMPUTE  GAMMA  INERTIAL   AND ITS TRIG FUNCTIONS      PDBC
    60 IF( AND(MM,MSK2).EQ.0) GO TO 70                                       PDBC
       SNGI = V*SINGAM/VI                                                    PDBC
       GAMI = ASIN(SNGI)                                                     PDBC
       CSGI = SQRT(1.- SNGI*SNGI)                                            PDBC
       ISKP =1                                                               PDBC
       IF(MM.LT.MSK3)GO TO 230                                               PDBC
C         TEST AND  COMPUTE   PSI INERTIAL  AND  ITS  TRIG  FUNCTIONS        PDBC
    70 IF( AND(MM,MSK3).EQ.0) GO TO 80                                       PDBC
       SPSII=(V*COSGAM*SINPSI+VE)/(VI *CSGI)                                 PDBC
       PSII = ATAN2(V*COSGAM*SINPSI+VE    ,V*COSGAM*COSPSI)                  PDBC
       CPSII= COS(PSII)                                                      PDBC
       ISKP =2                                                               PDBC
       IF(MM.LT.MSK4) GO TO 230                                              PDBC
C          TEST AND COMPUTE   MU INERTIAL                                    PDBC
    80 IF( AND(MM,MSK4).EQ.0) GO TO 90                                       PDBC
       XMUI= MU+ OMGZ*TIME                                                   PDBC
       IF(KK.LT.0.AND IPFLG1.NE.0) GO TO 540                                 PH1SZ
       IF(MM LT.MSK5) GO  TO 230                                             PDBC
C          TEST  AND  COMPUTE   SEMI-LATUS  RECTUM                           PDBC
    90 IF(AND(MM,MSK5).EQ 0) GO  TO  100                                     PDBC
       P= R*R* VI*VI* CSGI*CSGI /GM                                          PDBC
       IF(MM LT.MSK6) GO TO 230                                              PDBC
C          TEST  AND  COMPUTE  ECCENTRICITY                                  PDBC
   100 IF(AND(MM,MSK6) EQ.0) GO TO 110                                       PDBC
       RV2OMU =  R*VI*VI/GM                                                  PDBC
       ECC=  SQRT( 1. - RV2OMU*(2.-RV2OMU)*CSGI*CSGI )                       PDBC
       IF(MM LT.MSK7) GO TO 230                                              PDBC
C          TEST  AND  COMPUTE  INCLINATION   AND ITS  TRIG  FUNCTIONS        PDBC
   110 IF(AND(MM,MSK7).EQ.0) GO TO 120                                       PDBC
       CSI=  COSRHO* SPSII                                                   PDBC
       AINCL = ACOS(CSI)                                                     PDBC
       SNI =  SIN(AINCL)                                                     PDBC
       IF(MM.LT.MSK8) GO TO 230                                              PDBC
C         TEST  AND  COMPUTE  ARG. OF  PERIGEE                               PDBC
   120 IF(AND(MM,MSK8).EQ.0) GO TO  130                                      PDBC
```



```
      CSGNU =(P -R)/R/ECC                                              PDBC
      IF(ABS(CSGNU).GT.1 ) CSGNU=SIGN(1.,CSGNU)                        PH1SZ
      GNU = ACOS(CSGNU)                                                PDBC
      SRDI = SINRHO/SNI                                                PD14
      IF(ABS(SRDI).GT.1 )  SRDI = SIGN(1.,SRDI)                        PD14
      ARGP = ASIN( SRDI) - GNU                                         PD14
      IF(MM LT.MSK9) GO  TO 230                                        PDBC
C        TEST  AND  COMPUTE  LONG. OF  ASCENDING  NODE                 PDBC
  130 IF(AND(MM,MSK9) EQ.0) GO TO 140                                  PDBC
      SPSI = SPSII*SINRHO/SNI                                          PD14
      IF(ABS(SPSI) GT.1.)  SPSI=  SIGN(1.,SPSI)                        PD14
      ASCNOD = XMUI - ASIN(SPSI)                                       PD14
      IF(MM LT.MSK10) GO TO 230                                        PDBC
C        TEST  AND  COMPUTE  SEMI-MAJOR AXIS                           PDBC
  140 IF(AND(MM,MSK10) EQ.0) GO TO 150                                 PDBC
      SMIMAJ=   R*GM /(2.*GM -R*VI*VI)                                 PDBC
      IF(MM LT MSK11) GO TO 230                                        PDBC
C        TEST  AND  COMPUTE  APOGEE                                    PDBC
  150 IF(AND(MM,MSK11) EQ.0) GO TO 160                                 PDBC
      APOGEE=  SMIMAJ*(1.+ECC)                                         PDBC
      IF(MM LT MSK12) GO TO 230                                        PDBC
C        TEST  AND  COMPUTE  PERIGEE                                   PDBC
  160 IF(AND(MM,MSK12).EQ.0)  GO TO 170                                PDBC
      PERGEE =  SMIMAJ*(1.-ECC)                                        PDBC
      IF(MM.LT MSK13) GO TO 230                                        PDBC
C        TEST  AND  COMPUTE  TRUE  ANOMOLY                             PDBC
  170 IF(AND(MM,MSK13) EQ 0) GO TO 180                                 PDBC
      ANOMLY =  ATAN2( SNGI/CSGI, 1.- R/ P )                           PDBC
      IF(MM LT.MSK14) GO TO 230                                        PDBC
C       TEST  AND  COMPUTE   CAP X                                     PDBC
  180 IF(AND(MM,MSK14).EQ.0) GO TO 190                                 PDBC
      CAPX =  P/ECC                                                    PDBC
      IF(MM.LT MSK15) GO TO 230                                        PDBC
C        TEST  AND   COMPUTE  CAPY                                     PDBC
  190 IF(AND(MM,MSK15).EQ 0) GO TO 200                                 PDBC
      CAPY =  P*ECC / SQRT( ECC*ECC -1)                                PDBC
      IF(MM LT MSK16) GO TO 230                                        PDBC
C       TEST  AND  COMPUTE  ASSYMPTOTE                                 PDBC
  200 IF(AND(MM,MSK16) EQ.0) GO TO 210                                 PDBC
      ASYMP = ACOS(1./ECC)                                             PDBC
      IF(MM LT MSK17) GO  TO 230                                       PDBC
C       TEST  AND  COMPUTE  ENERGY                                     PDBC
  210 IF(AND(MM,MSK17).EQ.0) GO TO  220                                PDBC
      ENERGY =    2 *GM / SMIMAJ                                       PDBC
      IF(MM LT MSK18) GO TO 230                                        PDBC
C       TEST  AND  COMPUTE  MOMENTUM                                   PDBC
  220 IF(AND(MM,MSK18) EQ.0) GO TO  240                                PDBC
       HMNTM=  R*VI*CSGI                                               PDBC
      IF(MM GE.MSK19) GO TO 240                                        PDBC
      IF(KK.LT 0) GO TO 540                                            PDBC
C                RETURN  REQ. ORB. PARA. THROUGH ARG. LIST             PDBC
  230 F =  ORBPRM(K)                                                   PDBC
      RETURN                                                           PDBC
C        TEST AND  COMPUTE  PARTIALS  OF  V  INERTIAL                  PDBC
  240 IF(AND(MM,MSK19).EQ.0) GO TO 250                                 PDBC
      DVIDV =(V+R*OCORHO*COSGAM*SINPSI)/VI                             PDBC
      SNGI=V*SINGAM/VI                                                 PDBC
      DVIDG =             -VE*SNGI*SINPSI                              PDBC
      DVIDH =             VE*(V * COSGAM* SINPSI+ R*OCORHO)/ (R*VI)    PDBC
      DVIDM =0.                                                        PDBC
      DVIDPS=             VE*V*COSGAM*COSPSI/ VI                       PDBC
      DVIDRO= -SINRHO*VE*(V*COSGAM*SINPSI+ VE) / (COSRHO*VI)           PDBC
      DVIDMU= 0                                                        PDBC
      IF(MM.LT MSK20) GO TO 420                                        PDBC
C        TEST AND COMPUTE  PARTIALS  OF   GAMMA  INERTIAL              PDBC
  250 IF(AND(MM,MSK20).EQ.0) GO TO 260                                 PDBC
      VICI = VI* CSGI                                                  PDBC
      TMQ= - V* SINGAM/(VI* VICI)                                      PDBC
      DGIDV = SINGAM/VICI + TMQ*DVIDV                                  PDBC
```

```
      DGIDG = V*COSGAM/VICI + TMQ*DVIDG                                 PDBC
      DGIDH = TMQ*DVIDH                                                 PDBC
      DGIDM=0                                                           PDBC
      DGIDPS= TMQ* DVIDPS                                               PDBC
      DGIDRO= TMQ* DVIDRO                                               PDBC
      DGIDMU=0.                                                         PDBC
      IF(MM LT MSK21) GO TO 420                                         PDBC
C          TEST AND  COMPUTE  PARTIALS  PSI  INERTIAL                   PDBC
  260 IF(AND(MM,MSK21) EQ 0) GO TO 270                                  PDBC
      TM1= CPSII*CPSII /(V*COSGAM*COSPSI)                               PDBC
      DPIDV=-VE* TM1/ V                                                 PDBC
      DPIDH= -DPIDV*V/R                                                 PDBC
      DPIDG= DPIDH*R*SINGAM/COSGAM                                      PDBC
      DPIDPS=                                                           PDBC
     1        CPSII*CPSII/( COSPSI*COSPSI )+SINPSI*VE *TM1 /COSPSI      PDBC
      DPIDRO= -R*OMGZ*SINRHO*TM1                                        PDBC
      DPIDMU= 0.                                                        PDBC
      DPIDM = 0.                                                        PDBC
      IF(MM LT.MSK22) GO TO 420                                         PDBC
C            TEST   AND  COMPUTE  PARTIALS  OF  MU  INERTIAL            PDBC
  270 IF(AND(MM,MSK22) EQ 0) GO TO 280                                  PDBC
      DMIDV =0                                                          PDBC
      DMIDG =0.                                                         PDBC
      DMIDH =0                                                          PDBC
      DMIDM =0.                                                         PDBC
      DMIDPS=0.                                                         PDBC
      DMIDRO=0                                                          PDBC
      DMIDMU=1.                                                         PDBC
      IF(MM.LT MSK23) GO TO 420                                         PDBC
C          TEST AND  COMPUTE PARTIALS  OF  SEMI-LATUS RECTUM            PDBC
  280 IF(AND(MM,MSK23).EQ 0) GO TO 290                                  PDBC
      TM2= 2.* P /VI                                                    PDBC
      TM3= 2.* P / R                                                    PDBC
      TM4=-2.*P/CSGI *SNGI                                              PDBC
      DPDV= TM2*DVIDV + TM4*DGIDV                                       PDBC
      DPDG= TM2*DVIDG + TM4*DGIDG                                       PDBC
      DPDH= TM3 + TM2*DVIDH + TM4*DGIDH                                 PDBC
      DPDM= 0.                                                          PDBC
      DPDPS= TM2*DVIDPS + TM4*DGIDPS                                    PDBC
      DPDRO= TM2*DVIDRO + TM4*DGIDRO                                    PDBC
      DPDMU= 0                                                          PDBC
      IF(MM LT MSK24) GO TO 420                                         PDBC
C         TEST   AND  COMPUTE  PARTIALS  OF  ECCENTRICITY               PDBC
  290 IF(AND(MM,MSK24).EQ 0) GO TO 300                                  PDBC
      TMM=  (RV2OMU -1.)*RV2OMU*CSGI*CSGI/ECC                           PDBC
      TM5= 2 *TMM/VI                                                    PDBC
      TM6 =  RV2OMU*(2  - RV2OMU) * CSGI* SNGI/ ECC                     PDBC
      TM7 = TMM/ R                                                      PDBC
      DECDV= TM5*DVIDV + TM6*DGIDV                                      PDBC
      DECDG= TM5*DVIDG + TM6*DGIDG                                      PDBC
      DECDH= TM7+ TM5*DVIDH + TM6*DGIDH                                 PDBC
      DECDM=0                                                           PDBC
      DECDPS=TM5*DVIDPS + TM6*DGIDPS                                    PDBC
      DECDRO=TM5*DVIDRO + TM6*DGIDRO                                    PDBC
      DECDMU= 0                                                         PDBC
      IF(MM LT.MSK25) GO TO 420                                         PDBC
C          TEST AND COMPUTE PARTIALS  OF  INCLINATION                   PDBC
  300 IF(AND(MM,MSK25).EQ.0) GO TO 310                                  PDBC
      TM8 = -COSRHO* CPSII /SNI                                         PDBC
      DIDV=  TM8*DPIDV                                                  PDBC
      DIDG=  TM8*DPIDG                                                  PDBC
      DIDH=  TM8*DPIDH                                                  PDBC
      DIDM=  0.                                                         PDBC
      DIDPS= TM8*DPIDPS                                                 PDBC
      DIDRO= SINRHO*SPSII /SNI + TM8*DPIDRO                             PDBC
      DIDMU= 0.                                                         PDBC
      IF(MM LT MSK26) GO TO 420                                         PDBC
C         TEST   AND  COMPUTE  PARTIALS  OF  ARG  OF PERIGEE            PDBC
  310 IF(AND(MM,MSK26).EQ 0) GO TO  320                                 PDBC
      CGNAR  =  COS(GNU +ARGP)                                          PDBC
      SNGNU = SIN(GNU)                                                  PDBC
      DGNDF=  -1 /(R*ECC*SNGNU)                                         PDBC
```

```
      DGNDR =  -P*DGNDP/ R                                         PDBC
      DGNDE =  -(P-R)/ECC *DGNDP                                   PDBC
      TM9 = - SINRHO*CSI/SNI/SNI/CGNAR                             PDBC
      DBEDV =-DGNDP*DPDV - DGNDE*DECDV + TM9*DIDV                  PDBC
      DBEDG =-DGNDP*DPDG - DGNDE*DECDG + TM9*DIDG                  PDBC
      DBEDH = -DGNDR -DGNDP*DPDH -DGNDE*DECDH + TM9*DIDH           PDBC
      DBEDM =0.                                                    PDBC
      DBEDRO= COSRHO/(SNI*CGNAR)-DGNDP*DPDRO -DGNDE*DECDRO +TM9*DIDRO PDBC
      DBEDPS= -DGNDP*DPDPS -DGNDE*DECDPS +TM9*DIDPS                PDBC
      DBEDMU= 0.                                                   PDBC
      IF(MM.LT.MSK27) GO TO 420                                    PDBC
C           TEST AND COMPUTE  PARTIALS OF  LONG.OF ASCENDING NODE  PDBC
  320 IF(AND(MM,MSK27).EQ.0) GO TO 330                             PDBC
      CONO =-COS(ASCNOD -MU- OMGZ*TIME)                            PDBC
      DNODPI = CPSII *SINRHO/(SNI*CONO)                            PDBC
      DNODRR = SPSII*COSRHO/(SNI*CONO)                             PDBC
      DNODI = - SPSII*SINRHO*CSI /(SNI*SNI*CONO)                   PDBC
      DNODV = DNODPI*DPIDV +DNODI*DIDV                             PDBC
      DNODG = DNODPI*DPIDG +DNODI*DIDG                             PDBC
      DNODH = DNODPI*DPIDH +DNODI*DIDH                             PDBC
      DNODM =0.                                                    PDBC
      DNODPS=DNODPI*DPIDPS +DNODI*DIDPS                            PDBC
      DNODRO= DNODRR +DNODPI*DPIDRO + DNODI*DIDRO                  PDBC
      DNODMU= 1.                                                   PDBC
      IF(MM.LT.MSK28) GO TO 420                                    PDBC
C     TEST AND COMPUTE  PARTIALS OF  SMIMAJ                        PDBC
  330 IF(AND(MM,MSK28).EQ.0) GO TO 340                             PDBC
C     TEST  FOR   MOMENTUM                                         PDBC
      IF(K.EQ.36) GO TO 410                                        PDBC
      DSMDR =  SMIMAJ/R*(1. + SMIMAJ*VI*VI/GM)                     PDBC
      DSMDVI=  2.*SMIMAJ*SMIMAJ*VI /GM                             PDBC
      DSMDV = DSMDVI*DVIDV                                         PDBC
      DSMDG = DSMDVI*DVIDG                                         PDBC
      DSMDH- DSMDR + DSMDVI*DVIDH                                  PDBC
      DSMDM = 0.                                                   PDBC
      DSMDPS= DSMDVI* DVIDPS                                       PDBC
      DSMDRO= DSMDVI*DVIDRO                                        PDBC
      DSMDMU = 0.                                                  PDBC
C      TEST  FOR  ENERGY                                           PDBC
      IF(K.EQ.35) GO TO 400                                        PDBC
      IF(MM.LT.MSK29) GO TO 420                                    PDBC
C      TEST AND  COMPUTE PARTIALS OF APOGEE                        PDBC
  340 IF(AND(MM,MSK29).EQ.0) GO TO 350                             PDBC
      DAPDA = 1.+ECC                                               PDBC
      DAPDE =  SMIMAJ                                              PDBC
      DAPDV= DAPDA*DSMDV + DAPDE*DECDV                             PDBC
      DAPDG =DAPDA*DSMDG + DAPDE*DECDG                             PDBC
      DAPDH=DAPDA*DSMDH +DAPDE*DECDH                               PDBC
      DAPDM = 0.                                                   PDBC
      DAPDPS= DAPDA*DSMDPS + DAPDE*DECDPS                          PDBC
      DAPDRO= DAPDA*DSMDRO + DAPDE*DECDRO                          PDBC
      DAPDMU= 0.                                                   PDBC
      IF(MM.LT.MSK30) GO TO 420                                    PDBC
C       TEST AND COMPUTE  PARTIALS OF  PERIGEE                     PDBC
  350 IF(AND(MM,MSK30).EQ.0) GO TO  360                            PDBC
      DPEDA =  1.-ECC                                              PDBC
      DPEDE =  -SMIMAJ                                             PDBC
      DPEDV =  DPEDA*DSMDV + DPEDE*DECDV                           PDBC
      DPEDG =  DPEDA*DSMDG + DPEDE*DECDG                           PDBC
      DPEDH =  DPEDA*DSMDH + DPEDE*DECDH                           PDBC
      DPEDM = 0                                                    PDBC
      DPEDPS=  DPEDA*DSMDPS +DPEDE*DECDPS                          PDBC
      DPEDRO=  DPEDA*DSMDRO +DPEDE*DECDRO                          PDBC
      DPEDMU= 0.                                                   PDBC
      IF(MM.LT.MSK31) GO TO 420                                    PDBC
C       TEST  AND  COMPUTE PARTIALS OF  ANOMOLY                    PDBC
  360 IF(AND(MM,MSK31).EQ.0) GO TO 370                             PDBC
      CSANO =  COS(ANOMLY)                                         PDBC
      CS2  =  CSANO*CSANO                                          PDBC
      CON  =  1. - R/P                                             PDBC
      DANDGI = CS2/ CSGI/ CSGI/ CON                                PDBC
      DANDR  =  CS2*SNGI/(P*CON*CON*CSGI)                          PDBC
      DANDP  =  - DANDR * R/P                                      PDBC
```

572

```
              DANDV = DANDGI*DGIDV + DANDP*DPDV                                  PDBC
              DANDG = DANDGI*DGIDG + DANDP*DPDG                                  PDBC
              DANDH = DANDR + DANDGI*DGIDH + DANDP*DPDH                          PDBC
              DANDM =0.                                                          PDBC
              DANDPS = DANDGI*DGIDPS + DANDP*DPDPS                               PDBC
              DANDRO = DANDGI*DGIDRO + DANDP*DPDRO                               PDBC
              DANDMU = 0.                                                        PDBC
              IF(MM.LT.MSK32) GO TO 420                                          PDBC
      C          TEST   AND   COMPUTE  PARTIALS  OF  CAPX                        PDBC
          370 IF(AND(MM,MSK32).EQ.0) GO TO  380                                  PDBC
              DCXDP  = 1./ ECC                                                   PDBC
              DCXDE  = -P/(ECC*ECC)                                              PDBC
              DCXDV = DCXDP*DPDV + DCXDE*DECDV                                   PDBC
              DCXDG = DCXDP*DPDG + DCXDE*DECDG                                   PDBC
              DCXDH = DCXDP*DPDH + DCXDE*DECDH                                   PDBC
              DCXDM =0.                                                          PDBC
              DCXDPS= DCXDP*DPDPS +DCXDE*DECDPS                                  PDBC
              DCXDRO= DCXDP*DPDRO +DCXDE*DECDRO                                  PDBC
              DCXDMU= 0.                                                         PDBC
              IF(MM.LT.MSK33) GO TO 420                                          PDBC
      C          TEST  AND   COMPUTE  PARTIALS  OF    CAPY                       PDBC
          380 IF(AND(MM,MSK33).EQ.0)  GO  TO 390                                 PDBC
              TM1    =         ECC*ECC                                           PDBC
              TM2    =  TM1-1                                                    PDBC
              SQE2   =  SQRT(TM2)                                                PDBC
              DCYDP = ECC/ SQE2                                                  PDBC
              DCYDE =  P/SQE2*( 1. - TM1/ TM2)                                   PDBC
              DCYDV =DCYDP*DPDV + DCYDE*DECDV                                    PDBC
              DCYDG =DCYDP *DPDG +DCYDE*DECDG                                    PDBC
              DCYDH =DCYDP*DPDH + DCYDE*DECDH                                    PDBC
              DCYDM =0                                                           PDBC
              DCYDPS =DCYDP*DPDPS +DCYDE* DECDPS                                 PDBC
              DCYDRO = DCYDP*DPDRO +DCYDE* DECDRO                                PDBC
              DCYDMU = 0.                                                        PDBC
              IF(MM.LT.MSK34) GO TO 420                                          PDBC
      C          TEST  AND  COMPUTE  PARTIALS  OF  ASSYMPTOTE                    PDBC
          390 IF(AND(MM,MSK34).EQ. 0)  GO TO 420                                 PDBC
              SNASY = SIN(ASYMP)                                                 PDBC
              DASDE = 1./ ( SNASY*ECC*ECC)                                       PDBC
              DASDV =  DASDE*DECDV                                               PDBC
              DASDG =  DASDE*DECDG                                               PDBC
              DASDH =  DASDE*DECDH                                               PDBC
              DASDM = 0.                                                         PDBC
              DASDPS=  DASDE*DECDPS                                              PDBC
              DASDRO = DASDE* DECDRO                                             PDBC
              DASDMU =  0.                                                       PDBC
              GO TO  420                                                         PDBC
      C              COMPUTE   PARTIALS OF  ENERGY                               PDBC
          400 CONTINUE                                                           PDBC
              DENDSM= -2.*GM / (SMIMAJ* SMIMAJ)                                  PDBC
              DENDV = DENDSM * DSMDV                                             PDBC
              DENDG = DENDSM * DSMDG                                             PDBC
              DENDH = DENDSM * DSMDH                                             PDBC
              DENDM = 0.                                                         PDBC
              DENDPS= DENDSM * DSMDPS                                            PDBC
              DENDRO= DENDSM * DSMDRO                                            PDBC
              DENDMU= 0.                                                         PDBC
              GO   TO  420                                                       PDBC
      C         COMPUTE  PARTIALS  OF  MOMENTUM                                  PDBC
          410 CONTINUE                                                           PDBC
              DMODR=HMNTM/R                                                      PDBC
              DMODVI=  HMNTM / VI                                                PDBC
              DMODGI= - R*VI * SNGI                                              PDBC
              DMODV =  DMODVI*DVIDV + DMODGI *DGIDV                              PDBC
              DMODG =  DMODVI*DVIDG+DMODGI*DGIDG                                 PDBC
              DMODH =  DMODR  + DMODVI*DVIDH +DMODGI*DGIDH                       PDBC
              DMODM =0.                                                          PDBC
              DMODPS =DMODVI*DVIDPS +          DMODGI*DGIDPS                     PDBC
              DMODRO =DMODVI*DVIDRO +          DMODGI*DGIDRO                     PDBC
              DMODMU =0.                                                         PDBC
      C           STORE  PARTIALS IN  S  AND COMPUTE  SD                         PDBC
          420 K=K-18                                                             PDBC
```

20 OCT 72 6.01-46

```
      DO 430 J=1,7                                                    PDBC
      S(J) =PPO(J,K)                                                  PDBC
  430 CONTINUE                                                        PDBC
      IF(IOP.LT.2) RETURN                                             PDBC
      SD= 0.                                                          PDBC
      DO 440 J=1,7                                                    PDBC
      SD = SD + S(J)*F(J)                                             PDBC
  440 CONTINUE                                                        PDBC
      IF(K EQ 4.OR.K.EQ.9) SD=SD +OMGZ                                PDBC
      RETURN                                                          PDBC
C           IOP  =  2    RAPID ESTIMATE  OF TIME  DERIVATIVES         PDBC
  450 K=KK-11                                                         PDBC
      IF(K GT.9)CALL STPIT(43)                                        PDBC
      GO TO (460,470,480,490,500,510,510,510,510),K                   PDBC
  460 SD= VD                                                          PDBC
      GO TO 520                                                       PDBC
  470 SD= GD                                                          PDBC
      GO TO 520                                                       PDBC
  480 SD= PSID                                                        PDBC
      GO TO 520                                                       PDBC
  490 SD = UD+OMGZ                                                    PDBC
      GO TO 520                                                       PDBC
  500 SD = 2 *P/GM*(HD/R+VD/V -GD*SINGAM/COSGAM)                      PDBC
      GO TO 520                                                       PDBC
  510 SD = 0                                                          PDBC
      CALL STPIT(44)                                                  PDBC
  520 RETURN                                                          PDBC
C        KK GE 30   REENTRY TYPE TARGETS                              PDBC
  530 K= KK-29                                                        PDBC
      IF(IOP EQ 2)  GO TO (550,590,610,640,700,760,810),K             PDBC
      GO TO (540,570,600,630,690,750,800)                             PDBC
     * ,K                                                             PDBC
C          DOWNRANGE  CALCULATION                                     PDBC
  540 COSDMU = COS(MU-YMXRF)                                          PDBC
      SINDMU = SIN(MU-YMXRF)                                          PDBC
      TDNUM= CSPSR*(SINRHO*CSXLMR-COSRHO*SNXLMR*COSDMU)+SNPSR*COSRHO*SIN PDBC
     1DMU                                                             PDBC
      TDDEN= SINRHO*SNXLMR +COSRHO*CSXLMR*COSDMU                      PDBC
      TD = ATAN2(TDNUM,TDDEN)                                         PDBC
      SDOWN =TD*ER                                                    PDBC
      IF(KK LT 0) GO TO 580                                           PDBC
      IF(IOP GT.0) GO TO 550                                          PDBC
      F= TD*ER                                                        PDBC
      GO TO 560                                                       PDBC
C             PARTIALS  OF  DOWNRANGE                                 PDBC
  550 COSTD = COS(TD)                                                 PDBC
      S(1)  =0                                                        PDBC
      S(2)  =0                                                        PDBC
      S(3)  =0                                                        PDBC
      S(4)  =0.                                                       PDBC
      S(5)  =0.                                                       PDBC
      S(6)  =ER*COSTD*COSTD*((CSPSR*(COSRHO*CSXLMR+SINRHO*SNXLMR*COSDMU) PDBC
     1   -SNPSR*SINRHO*SINDMU)/TDDEN - TDNUM*(COSRHO*SNXLMR -SINRHO*CSXL PDBC
     2MR +  COSDMU)/(TDDEN*TDDEN) )                                   PDBC
      S(7)   =ER*COSTD*COSTD*(( CSPSR*COSRHO*SNXLMR*SINDMU +SNPSR*COSRHO PDBC
     1*COSDMU)/TDDEN  + TDNUM*COSRHO*CSXLMR*SINDMU/(TDDEN*TDDEN) )    PDBC
      IF(IOP EQ.1) RETURN                                             PDBC
      SD =  S(6)*F(6)  + S(7)*F(7)                                    PDBC
  560 RETURN                                                          PDBC
C            CROSS  RANGE  COMPUTATION                                PDBC
  570 COSDMU = COS(MU-YMXRF)                                          PDBC
      SINDMU = SIN(MU-YMXRF)                                          PDBC
  580 CONTINUE                                                        PDBC
      STC = CSPSR *COSRHO*SINDMU - SNPSR*(SINRHO*CSXLMR -COSRHO*SNXLMR*  PDBC
     1         COSDMU)                                                PDBC
      TC = ASIN(STC)                                                  PDBC
      SCROSS=TC*ER                                                    PDBC
      IF(KK LT 0) GO TO 610                                           PDBC
```

O OCT 72 G.01-46

```
      IF(IOP.GT 0) GO TO 590                                              PDBC
      F=TC*ER                                                             PDBC
      GO TO 560                                                           PDBC
C          PARTIALS   OF    CROSS RANGE                                   PDBC
  590 COSTC= COS(TC)                                                      PDBC
      S(1)=0                                                              PDBC
      S(2)=0.                                                             PDBC
      S(3)=0.                                                             PDBC
      S(4)=0.                                                             PDBC
      S(5)=0.                                                             PDBC
      S(6)= ER/COSTC*(-CSPSR*SINRHO*SINDMU - SNPSR*(COSRHO*CSXLMR+SINRHO  PDBC
     1*   SNXLMR* COSDMU))                                                PDBC
      S(7) = ER/COSTC*(CSPSR*COSRHO*COSDMU - SNPSR*COSRHO*SNXLMR*SINDMU)  PDBC
      IF(IOP EQ 1) RETURN                                                 PDBC
      SD =  S(6)*F(6) + S(7)*F(7)                                         PDBC
      RETURN                                                              PDBC
C        TOTAL RANGE                                                      PDBC
  600 COSDMU = COS(MU-YMXRF)                                              PDBC
  610 CONTINUE                                                            PDBC
      COSTHT = SINRHO* SNXLMR  + COSRHO*CSXLMR * COSDMU                   PDBC
      THT  =  ACOS(COSTHT)                                                PDBC
      STOT = ER*THT                                                       PDBC
      IF(KK.LT.0) RETURN                                                  PDBC
      IF(IOP GT.0) GO TO 620                                              PDBC
      F= ER*THT                                                           PDBC
      GO TO 560                                                           PDBC
  620 SNTHT = SIN(THT)                                                    PDBC
      S(1)=0.                                                             PDBC
      S(2)=0.                                                             PDBC
      S(3)=0.                                                             PDBC
      S(4)=0                                                              PDBC
      S(5)=0                                                              PDBC
      S(6)= -ER*(COSRHO*SNXLMR -SINRHO*CSXLMR*COSDMU) / SNTHT             PDBC
      SINDMU= SIN(MU-YMXRF)                                               PDBC
      S(7)=  COSRHO*CSXLMR*SINDMU / SNTHT                                 PDBC
     *  *ER                                                               PDBC
      IF(IOP EQ 1) GO TO 560                                              PDBC
      SD  =  S(6)*F(6) + S(7)*F(7)                                        PDBC
      GO TO  560                                                          PDBC
C          DYNAMIC  PRESSURE                                              PDBC
  630 IF(IOP NE.0) GO TO 640                                              PDBC
      F= Q                                                                PDBC
      RETURN                                                              PDBC
C          PARTIALS  OF  DYNAMIC PRESSURE                                 PDBC
  640 S(1) = 2.*Q/V                                                       PDBC
      S(2)=0                                                              PDBC
      IF(IATM-1) 660,670,650                                              PDBC
  650 CALL STPIT(46)                                                      PDBC
  660 CALL ANLATM(ALT,PA,1)                                               PDBC
      GO TO 680                                                           PDBC
  670 CALL PAT63(ALT,PA,2)                                                PDBC
  680 S(3)=Q/RO*ROR                                                       PDBC
      S(4)=0.                                                             PDBC
      S(5)=0.                                                             PDBC
      S(6)=0.                                                             PDBC
      S(7)=0                                                              PDBC
      IF(IOP EQ.1) RETURN                                                 PDBC
      SD = F(1)*S(1)+ F(3)* S(3)                                          PDBC
      RETURN                                                              PDBC
C          HEATING  RATE                                                  PDBC
  690 IF(IOP.GT 0) GO TO  700                                             PDBC
      F = HTD                                                             PDBC
      RETURN                                                              PDBC
C           HEATING  RATE  PARTIALS                                       PDBC
  700 S(1) =   3.15* HTD/V                                                PDBC
      S(2) =0.                                                            PDBC
      IF(IATM -1) 710,720,650                                             PDBC
  710 CALL ANLATM(ALT,PA,1)                                               PDBC
      GO TO 730                                                           PDBC
```

0 OCT 72 G 01-46

```
      720 CALL  PAT63(ALT,PA,2)                                      PDBC
      730 S(3) =  5 * HTD*ROR/RO                                     PDBC
      740 S(4) =0                                                    PDBC
          S(5)= 0.                                                   PDBC
          S(6)= 0.                                                   PDBC
          S(7)= 0.                                                   PDBC
          IF(IOP.EQ.1) RETURN                                        PDBC
          SD =  S(1)*F(1) + S(3)*F(3)                                PDBC
          RETURN                                                     PDBC
    C       REYNOLDS NUMBER                                          PDBC
      750 RE  = V/VNU                                                PDBC
          IF(IOP GT.0) GO TO  760                                    PDBC
          F = RE                                                     PDBC
          RETURN                                                     PDBC
    C           PARTIALS  OF  REYNOLS  NUMBER                        PDBC
      760 S(1)  = 1./ VNU                                            PDBC
          S(2)  = 0.                                                 PDBC
          IF(IATM-1) 770,780,650                                     PDBC
      770 CALL ANLATM(ALT,PA,1)                                      PDBC
          GO TO  790                                                 PDBC
      780 CALL PAT63(ALT,PA,2)                                       PDBC
      790 S(3) = - RE/VNU*VNR                                        PDBC
          GO TO  740                                                 PDBC
    C              PAYLOAD WEIGHT                                    DS
      800 CALL WTDRP(SQ(3,5)-W,ORBDRP,DW2,-1)                        JULY28
          CALL PAYLOD(PLOAD,DP2,SQ(4,1),SQ(3,5),W)                   JULY28
          IF(IOP.GT.0) GO TO 810                                     DS
          F= PLOAD                                                   DS
          RETURN                                                     DS
    C          PAYLOAD PARTIALS                                      DS
      810 S(1)=0.                                                    PDBC
          S(2)=0.                                                    PDBC
          S(3)=0.                                                    PDBC
          S(4)= GR*DP2                                               JULY28
          S(5)=0.                                                    PDBC
          S(6)=0                                                     PDBC
          S(6)=0.                                                    PDBC
          S(7)=0                                                     PDBC
          IF(IOP.GT 1) SD = S(4)*F(4)                                DS
          RETURN                                                     PDBC
          END                                                        PDBC
```

SUBRØUTINE
PDY3A

Subroutine PDY3A

Entry PDY

Purpose

PDY or PDY3A combines and calculates complete partial derivatives of the equations of motion with respect to the state for the adjoint differential equation coefficients. PDY is called from MODEL (MODELB).

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
AG	a^{γ}	I	Acceleration vector element.	/AXL /(	2)	ACCEL	M	AG
						DER3A	I	AG
						PDY3A	I	AG
AGM		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	30)	ACCEL	O	AGM
						PDY3A	I	AGM
AGR		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	18)	ACCEL	O	AGR
						PDY3A	I	AGR
AGV		I	Element of matrix of acceleration vector partials WRT state	/AXL /(	6)	ACCEL	O	AGV
						PDY3A	I	AGV
AMM		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	32)	PDY3A	I	AMM
AMR		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	20)	ACCEL	O	AMR
						PDY3A	I	AMR
AMV		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	8)	ACCEL	O	AMV
						PDY3A	I	AMV
AP	a^{ψ}	I	Acceleration vector element.	/AXL /(	3)	ACCEL	M	AP
						DER3A	I	AP
						PDY3A	I	AP
APM		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	31)	ACCEL	O	APM
						PDY3A	I	APM
APR		I	Element of matrix of acceleration vector partials WRT state	/AXL /(	19)	ACCEL	O	APR
						PDY3A	I	APR
APV		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	7)	ACCEL	O	APV
						PDY3A	I	APV
AVM		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	29)	ACCEL	O	AVM
						PDY3A	I	AVM
AVR		I	Element of matrix of acceleration vector partials WRT state.	/AXL /(	17)	ACCEL	O	AVR
						PDY3A	I	AVR
AVV		I	Name of acceleration partials matrix.	/AXL /(	5)	ACCEL	M	AVV
						ACCEL	O	AY
						PDY3A	I	AVV
COSGAM	$\cos(\gamma)$	I	See symbol	/STATE3/(	687)	ACCEL	I	COSGAM
						BL4	I	COSGAM
						BL8	I	COSGAM
						DER3A	I	COSGAM
						EQUA3	O	COSGAM
						MODELA	I	COSGAM
						MODELB	I	COSGAM
						OUT	I	COSGAM
						PDBC	I	COSGAM
						PDY3A	I	COSGAM
COSPSI	$\cos(\psi)$	I	See symbol	/STATE3/(	705)	BL4	I	COSPSI
						BL7	I	COSPSI
						BL8	I	COSPSI
						DER3A	I	COSPSI
						EQUA3	O	COSPSI
						MODELA	I	COSPSI
						MODELB	I	COSPSI
						PDBC	I	COSPSI
						PDY3A	I	COSPSI
COSRHO	$\cos(\rho)$	I	See symbol	/STATE3/(	707)	BL4	I	COSRHO
						BL7	I	COSRHO
						BL8	I	COSRHO
						DER3A	I	COSRHO
						EQUA3	M	COSRHO
						MODELA	I	COSRHO
						MODELB	I	COSRHO
						OUT	I	COSRHO
						PDBC	I	COSRHO
						PDY3A	I	COSRHO
DGDH		I	Partial of gravity wrt altitude	/GENF /(	563)	BL7	I	GH
						BL8	I	GH
						EQUA3	O	GH
						PDY3A	I	DGDH

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
G	g	I	Gravitational attraction (FT/SEC**2)	/GENF /(	301)	BL4	I	G
						BL7	I	G
						BL8	I	G
						DER3A	I	G
						EQUA3	M	G
						MODELA	I	G
						MODELB	I	G
						PDY3A	I	G
						SDER3	I	G
						SDINP	M	G
GDG		O	Partial derivative of equation of motion WRT state	/STATE3/(	729)	ADEQ3A	I	GDG
						PDY3A	O	GDG
GDM		O	Partial derivative of equation of motion WRT state	/STATE3/(	741)	ADEQ3A	I	GDM
						PDY3A	O	GDM
GDO		O	Partial derivative of equation of motion WRT state	/STATE3/(	750)	ADEQ3A	I	GDO
						PDY3A	O	GDO
GDP		O	Partial derivative of equation of motion WRT state	/STATE3/(	745)	ADEQ3A	I	GDP
						PDY3A	O	GDP
GDPH	$\partial\dot{\gamma}/\partial\phi$	O	See symbol	/AECO3 /(	14)	ACCEL	O	GDPH
						ADEQ3A	I	GDPH
						PDY3A	O	GDPH
GDR		O	Partial derivative of equation of motion WRT state	/STATE3/(	735)	ADEQ3A	I	GDR
						PDY3A	O	GDR
GDV		O	Partial derivative of equation of motion WRT state	/STATE3/(	722)	ADEQ3A	I	GDV
						PDY3A	O	GDV
HTD	$\dot{Q}$	M	Heating derivative	/STATE3/(	22)	DER3A	O	HTD
						OUT	I	HTD
						PDBC	I	HTD
						PDY3A	M	HTD
HTDR		O	Partial derivative of equation of motion WRT state	/STATE3/(	754)	ADEQ3A	I	HTDR
						PDY3A	O	HTDR
HTDV		O	Partial derivative of equation of motion WRT state	/STATE3/(	753)	ADEQ3A	I	HTDV
						PDY3A	O	HTDV
JGII		I	Control option	/XCODES/(	195)	ACCEL	I	JGII
						BNTG	O	JGII
						DER3A	I	JGII
						FNTG	M	JGII
						GUI3A	I	JGII
						MODELA	I	JGII
						MODELB	I	JGII
						MTX3A	I	JGII
						PDY3A	I	JGII
JPRP		I	Propulsion flag for different rocket options	/XCODES/(	194)	ACCEL	I	JPRP
						DER3A	I	JPRP
						EQUA3	I	JPRP
						MODELA	I	JPRP
						PDY3A	I	JPRP
						PROPB	O	JPRP
						PROPIN	O	JPRP
MDM		O	Partial derivative of equation of motion WRT state	/STATE3/(	742)	ADEQ3A	I	MDM
						PDY3A	O	MDM
MDR		O	Partial derivative of equation of motion WRT state	/STATE3/(	736)	ADEQ3A	I	MDR
						PDY3A	O	MDR
MDV		O	Partial derivative of equation of motion WRT state	/STATE3/(	724)	ADEQ3A	I	MDV
						PDY3A	O	MDV
OCORHO	$\omega\times$COSRHO	I	See symbol	/STATE3/(	708)	DER3A	I	OCORHO
						EQUA3	M	OCORHO
						MODELA	I	OCORHO
						MODELB	I	OCORHO
						PDBC	I	OCORHO
						PDY3A	I	OCORHO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
OCORO2	$\omega \times$OCORHO	I	See symbol	/STATE3/(	709)	DER3A EQUA3 MODELA MODELB PDY3A	I O I I I	OCORO2 OCORO2 OCORO2 OCORO2 OCORO2
ODG		O	Partial derivative of equation of motion WRT state	/STATE3/(	732)	ADEQ3A PDY3A	I O	ODG ODG
ODP		O	Partial derivative of equation of motion WRT state	/STATE3/(	747)	ADEQ3A PDY3A	I O	ODP ODP
ODR		O	Partial derivative of equation of motion WRT state	/STATE3/(	738)	ADEQ3A PDY3A	I O	ODR ODR
ODV		M	Partial derivative of equation of motion WRT state	/STATE3/(	726)	ADEQ3A PDY3A	I M	ODV ODV
OMGZ	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A CRASH DER3A EQUA3 GEINP MODELA MODELB PDBC PDY3A SDINP TOPM	I I I I I I I I I I I	OMGZ OMEGA OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ OMGZ
PDG		O	Partial derivative of equation of motion WRT state	/STATE3/(	731)	ADEQ3A PDY3A	I O	PDG PDG
PDM		O	Partial derivative of equation of motion WRT state	/STATE3/(	743)	ADEQ3A PDY3A	I O	PDM PDM
PDO		O	Partial derivative of equation of motion WRT state	/STATE3/(	751)	ADEQ3A PDY3A	I O	PDO PDO
PDP		O	Partial derivative of equation of motion WRT state	/STATE3/(	746)	ADEQ3A PDY3A	I O	PDP PDP
PDPH	$\partial\dot{\psi}/\partial\phi$	O	See symbol	/AECO3 /(	15)	ACCEL ADEQ3A PDY3A	O I O	PDPH PDPH PDPH
PDR		O	Partial derivative of equation of motion WRT state	/STATE3/(	737)	ADEQ3A PDY3A	I O	PDR PDR
PDV		O	Partial derivative of equation of motion WRT state	/STATE3/(	725)	ADEQ3A PDY3A	I O	PDV PDV
QMULT	=0 OR 1	I	Heating flag multiplier	/ARCDAT/(	40)	DER3A FXDAT PDY3A	I O I	QMULT QMULT QMULT
R	R	I	Radial distance from earth center to vehicle (FT)	/GENF /(	305)	BL4 BL7 BL8 DER3A EQUA3 MODELA MODELB PDBC PDY3A TRTOSZ	I I I I M I I I I I	R R R R R R R R R R
RDG		O	Partial derivative of equation of motion WRT state	/STATE3/(	730)	ADEQ3A PDY3A	I O	RDG RDG
RDV		O	Partial derivative of equation of motion WRT state	/STATE3/(	723)	ADEQ3A PDY3A	I O	RDV RDV
RHOB	ρ_b	I	Atmosphere base density for heating calculation (LB/FT**3)	/ARCDAT/(	39)	DER3A FXDAT PDY3A	I O I	RHOB RHOB RHOB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
RO	ρ_a	I	Atmospheric density (SLUGS/FT**3)	/GENF /(	309)	BL7	I	RO
						BL8	I	RO
						DER3A	I	RO
						EQUA3	I	RO
						OUT	I	RO
						PDBC	I	RO
						PDY3A	I	RO
ROR		I	Deriv Of density wrt alt.	/GENF /(	313)	BL7	I	ROR
						BL8	I	ROR
						EQUA3	I	ROR
						PDBC	I	ROR
						PDY3A	I	ROR
SINGAM	$\sin(\gamma)$	I	See symbol	/STATE3/(	688)	BL4	I	SINGAM
						BL7	I	SINGAM
						BL8	I	SINGAM
						DER3A	I	SINGAM
						EQUA3	O	SINGAM
						MODELA	I	SINGAM
						MODELB	I	SINGAM
						PDBC	I	SINGAM
						PDY3A	I	SINGAM
						SDER3	I	SINGAM
SINPSI	$\sin(\psi)$	I	See symbol	/STATE3/(	704)	BL4	I	SINPSI
						BL7	I	SINPSI
						BL8	I	SINPSI
						DER3A	I	SINPSI
						EQUA3	O	SINPSI
						MODELA	I	SINPSI
						MODELB	I	SINPSI
						PDBC	I	SINPSI
						PDY3A	I	SINPSI
SINRHO	$\sin(\rho)$	I	See symbol	/STATE3/(	706)	BL4	I	SINRHO
						BL7	I	SINRHO
						BL8	I	SINRHO
						DER3A	I	SINRHO
						EQUA3	O	SINRHO
						MODELA	I	SINRHO
						MODELB	I	SINRHO
						OUT	I	SINRHO
						PDBC	I	SINRHO
						PDY3A	I	SINRHO
SQRT		F	Square root function	/SQRT /(S	)	ANLATM	F	SQRT
						CRASH	F	SQRT
						DCTDE	F	SQRT
						DER3A	F	SQRT
						ENVPRM	F	SQRT
						HUNT	F	SQRT
						MODELA	F	SQRT
						MODELB	F	SQRT
						OPWELL	F	SQRT
						OUT	F	SQRT
						PAT63	F	SQRT
						PAY02	F	SQRT
						PDBC	F	SQRT
						PDY3A	F	SQRT
						STORE	F	SQRT
						SYMVRT	F	SQRT
						WTSCH	F	SQRT
UDG		O	Partial derivative of equation of motion WRT state	/STATE3/(	733)	ADEQ3A	I	UDG
						PDY3A	O	UDG
UDO		O	Partial derivative of equation of motion WRT state	/STATE3/(	752)	ADEQ3A	I	UDO
						PDY3A	O	UDO
UDP		O	Partial derivative of equation of motion WRT state	/STATE3/(	748)	ADEQ3A	I	UDP
						PDY3A	O	UDP
UDR		M	Partial derivative of equation of motion WRT state	/STATE3/(	739)	ADEQ3A	I	UDR
						PDY3A	M	UDR
UDV		M	Partial derivative of equation of motion WRT state	/STATE3/(	727)	ADEQ3A	I	UDV
						PDY3A	M	UDV

30 OCT 72 G.01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
V	v	I	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						ASI3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	D	KWOW
						YREF3	M	V
VDG		O	Partial derivative of equation of motion WRT state	/STATE3/(	728)	ADEQ3A	I	VDG
						PDY3A	O	VDG
VDM		O	Partial derivative of equation of motion WRT state	/STATE3/(	740)	ADEQ3A	I	VDM
						PDY3A	O	VDM
VDO		O	Partial derivative of equation of motion WRT state	/STATE3/(	749)	ADEQ3A	I	VDO
						PDY3A	O	VDO
VDP		O	Partial derivative of equation of motion WRT state	/STATE3/(	744)	ADEQ3A	I	VDP
						PDY3A	O	VDP
VDR		O	Partial derivative of equation of motion WRT state	/STATE3/(	734)	ADEQ3A	I	VDR
						PDY3A	O	VDR
VDV		O	Partial derivative of equation of motion WRT state	/STATE3/(	721)	ADEQ3A	I	VDV
						PDY3A	O	VDV

```
      SUBROUTINE PDY3A                                                     PDY3B
C                                                                          COMN
C         COMBINE  AND  CALCULATE   PARTIALS  FOR  ADJOINTS                COMN
C                                                                          COMN
      COMMON/ARCDAT/                                                       ARCDAT
     *SREF       ,EJ         ,XISP       ,TMULT      ,DTNC       ,DTPI      , ARCDAT
     *IATM       ,IMODE      ,JAER       ,JPRD       ,DMAX       ,SMAX      , ARCDAT
     *XLMAX      ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX      ,MAEA      , ARCDAT
     *MAEB       ,MAEC       ,MAED       ,MAEE       ,MAEF       ,MAEG      , ARCDAT
     *MT         ,MISP       ,MXCG       ,MZCG       ,MWDA       ,MWDB      , ARCDAT
     *MDB        ,XCGR       ,ZCGR       ,XE         ,ZE         ,XT        , ARCDAT
     *DREF       ,MCND       ,RHOB       ,QMULT      ,REMAX                   ARCDAT
     *  ,FRATE  , ARCD(9)                                                  RETAP
      DIMENSION  ARCDA(40)                                                 ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                              ARCDAT
      COMMON/STATE3/                                                       STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,Y0(9)      ,SVY(10)   , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI     ,SINRHO     ,COSRHO     ,DCORHO     ,DCORO2    , STATE3D
     *SVBV   (9),OMEGA       ,OMEGA2     ,                                   STATE3D
     *VDV        ,GDV        ,RDV        ,MDV        ,PDV        ,ODV       , STATE3D
     *UDV        ,VDG        ,GDG        ,RDG        ,PDG        ,ODG       , STATE3D
     *UDG        ,VDR        ,GDR        ,MDR        ,PDR        ,ODR       , STATE3D
     *UDR        ,VDM        ,GDM        ,MDM        ,PDM        ,VOP       , STATE3D
     *GDP        ,PDP        ,ODP        ,UDP        ,VDO        ,GDO       , STATE3D
     *PDO        ,UDO        ,HTDV       ,HTDR                                STATE3D
      REAL MDM , MDV, MDR                                                  STATE3D
      COMMON/STATE3/                                                       STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                       STATE3D
      COMMON/AECO3/                                                        AECO3
     *APHO       ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA       , AECO3
     *SINA       ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI    , AECO3
     *COSPHI     ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDO       , AECO3
     *CDOM       ,CLD        ,FK         ,XCGM       ,ZCGM       ,CLOM      , AECO3
     *CM         ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM       ,FKM       , AECO3
     *CLAM       ,CL         ,CLA        ,CLM        ,                        AECO3
     *CD         ,CDA        ,CDM                                          AECO3
      COMMON/GENF/                                                         GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)    ,TOL(9)     ,SVAR(10)   ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)    ,DTP       , GENF
     *DTS        ,DT         ,G          ,DPSQ       ,Q          ,QS        , GENF
     *R          ,RE         ,MACH       ,PA         ,RO         ,CS        , GENF
     *VNU        ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)    ,DIP(20)    ,T          ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)   ,DIS1(20)   ,TIME       ,OMP       , GENF
     *TIMPR      ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)   , GENF
     *AE         ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV     , GENF
     *QR         ,QV         ,FVAC       ,LIFTV      ,                        GENF
     *LIFTR      ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA     , GENF
     *            LIFTM      ,DBR        ,DB         ,ISP        ,ISPF      , GENF
     *                        ULFT       ,ULFTV      ,ULFTR      ,ULFTA     , GENF
     *XMCG       ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE     , GENF
     *CULFT      ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SID       , GENF
     *COD        ,SIDAE      ,XCG        ,ZCG        ,XJ                      GENF
      COMMON / GENF  /                                                     GENF
     *XJV         XJR        ,GH         ,GAMMAD     ,XKG        ,XKP       , GENF
     *FRATED     ,IRATED     ,                                              GENF
     *P1         ,P2         ,P3        ,XK1         ,XK2        ,XK3       , GENF
     *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D      , GENF
     *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V      , GENF
     *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P      , GENF
     *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O      , GENF
     *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M      , GENF
     *PV         ,PG         ,PP         ,PR         ,PD         ,OPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                     FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      COMMON /XCODES/                                                      XCODES
     *ITQ    (9),ICOR  (20),ITI         ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST      ,NSB        ,NSAB       ,NICNB     , XCODES
     *I20P       ,ICOP       ,IFAW       ,IFAR       ,IFB        ,IND       , XCODES
     *IOPEN      ,IPH        ,ISPH       ,ISST       ,IARC       ,ISTART    , XCODES
     *ITCT       ,ITER       ,IVAR       ,JK         ,JPS        ,JS        , XCODES
```

20 OCT 72 G 01-46

```
      *KOP        ,KPST       ,K          ,KST        ,NAD        ,NCASE      ,  XCODES
      *NCN        ,NEQB       ,NEQ        ,NOP        ,NPH        ,N          ,  XCODES
      *NST        ,IPST       ,IPRINT     ,ISTN       ,IPHN       ,ISTNB      ,  XCODES
      *IPHNB      ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP      ,L          ,  XCODES
      *IFOB       ,NB         ,LB         ,MB         ,NPHP       ,NPHB       ,  XCODES
      *NCTIN      ,NEQF  ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3             XCODES
       EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,          EQUV3
      *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2),         EQUV3
      *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),         EQUV3
      *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)                    EQUV3
       REAL M,MU,MD                                                              EQUV3
       COMMON/DATA/                                                              DATA
      *PI         ,RAD        ,RDI        ,SC         ,UMF        ,TMPF       ,  DATA
      *FTNM       ,CAR        ,JOP1       ,JOP2       ,JOP3       ,JOP4          DATA
       COMMON/GLOBAL/                                                            GLOBAL
      *GR         ,ER         ,OMGZ       ,XLAMRF     ,YMURF      ,LUM           GLOBAL
      *,JJOP(10)  ,IFATAL     ,NARC       ,NBRAN      ,NFARC      ,ID(4)         GLOBAL
      *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                                      GLOBAL
      *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)            GLOBAL
      *,ITPSO ,KSOL     ,KGLOBL(8)                                               RETAP
       COMMON/ AXL/                                                              AXL
      *AV    ,AG    ,AP    ,AM    ,AVV   ,AGV   ,APV   ,AMV   ,AVG   ,AGG   ,APG   ,  AXL
      *AMG   ,AVP   ,AGP   ,APP   ,AMP   ,AVR   ,AGR   ,APR   ,AMR   ,AVO   ,AGO   ,  AXL
      *APO   ,AMO   ,AVU   ,AGU   ,APU   ,AMU   ,AVM   ,AGM   ,APM   ,AMM            AXL
       DIMENSION  AY(1)                                                          AXL
       EQUIVALENCE (AY,AVV)                                                      AXL
       EQUIVALENCE(GH,OGDH)                                                      PO14
       ENTRY  PDY                                                                PDY3B
C         I     CALC.   SPEC.  IMPULSE  AND  PARTIALS                            COMN
       IF(JPRP GT.1)  CALL  IMPUL                                                COMN
C                                                                                COMN
C         II    CALC.  ACCELERATION VECTOR  AND  MATRIX OF PARTIALS              COMN
       CALL  ACCEL                                                               COMN
C                                                                                COMN
C        III  GRIND  OUT       PARTIAL DERIVATIVES                               COMN
       VSQ = V* V                                                                PDY3B
       VDV= AVV                                                                  PDY3B
       ROSQ =  R* OCORO2                                                         PDY3B
       VDG=  ROSQ* ( COSRHO*COSGAM+SINRHO*COSPSI*SINGAM) -G*COSGAM               PDY3B
       VDP=  ROSQ* SINRHO* SINPSI*COSGAM                                         PDY3B
       VDR= OCORO2*(COSRHO*SINGAM- SINRHO*COSPSI*COSGAM) - OGDH*SINGAM+AV        PDY3B
      1R                                                                         PDY3B
       VDO=  ROSQ *( -2.*SINRHO*SINGAM- COSRHO*COSPSI*COSGAM+ SINRHO*SINR        PDY3B
      1HO*COSPSI* COSGAM/COSRHO)                                                 PDY3B
       VDM= AVM                                                                  PDY3B
       IF(JGII EQ 8) GO TO 9                                                     PO14    9
       GDV= COSGAM*(1./R+ G/VSQ)  - ROSQ/VSQ*(COSRHO*COSGAM+SINRHO*COSPSI        PDY3B
      1*SINGAM)+AGV/V-AG/VSQ                                                     PDY3B
       GDG= SINGAM*(G/V-V/R) + ROSQ/V*( SINRHO*COSPSI*COSGAM-    COSRHO*S        PDY3B
      1INGAM)                                                                    PDY3B
       GDP= 2.*OMGZ*COSRHO*COSPSI- ROSQ/V*SINRHO*SINPSI*SINGAM                   PDY3B
       GDR= COSGAM*( -V/(R*R) -DGDH/V) +OCORO2/V*(COSRHO*COSGAM+   SINRHO        PDY3B
      1*COSPSI*SINGAM) + AGR/V                                                   PDY3B
       GDO= -2 *OMGZ*SINRHO*SINPSI+  ROSQ/V*( -2.*SINRHO*COSGAM+ COSPSI*         PDY3B
      1  SINGAM*( COSRHO- SINRHO*SINRHO/COSRHO))                                 PDY3B
       GDM= AGM/V                                                                PDY3B
       IF( ABS(COSGAM).LT 1.E-11) GO TO  10                                      PDY3B   10
       PDV= -AP/(VSQ*COSGAM)                                                     PDY3B
      1         + SINRHO*COSGAM*SINPSI/(R*COSRHO)                                PDY3B
      2      + APV/(V*COSGAM)                                                    PDY3B
      * -ROSQ*SINRHO*SINPSI/(VSQ*COSGAM)                                         PO14
       CSGSQ=COSGAM*COSGAM                                                       PDY3B
       PDG= OCORHO/CSGSQ*(R*OMGZ*SINRHO*SINPSI*SINGAM/V- 2 *COSPSI)              PDY3B
      1 -SINRHO*V*SINPSI*SINGAM/(R*COSRHO)                                       PDY3B
      * +  AP*SINGAM/(CSGSQ*V)                                                   MAR14
       PDP= OCORHO/COSGAM*(R*OMGZ*SINRHO*COSPSI/V+2.*SINPSI*SINGAM)              PDY3B
      1     +V*SINRHO*COSGAM*COSPSI/R/COSRHO                                     PDY3B
       PDR=  SINRHO* SINPSI*( OCORO2/(V*COSGAM) - V*COSGAM/(R*R*COSRHO))         PDY3B
      1 +APR/(V*COSGAM)                                                          PDY3B
       PDO=  V*COSGAM*SINPSI/(R*COSRHO*COSRHO)  +2 *OCORHO+                      PDY3B
      1 OMGZ/COSGAM*(2.*SINRHO*COSPSI*SINGAM+ R*OMGZ*SINPSI/V*(                  PDY3B
      2  COSRHO*COSRHO-SINRHO*SINRHO))                                           PDY3B
       PDM= APM/(V*COSGAM)                                                       PDY3B
       GO TO 20                                                                  PDY3B   20
C                                                                                COMN
```

515

```
      C        IV    PARTIALS  ON  VERTICAL  RISE                          COMN
            9 GOV=0.                                                        PD14
              GOG=0.                                                        PD14
              GOP=0.                                                        PD14
              GOR=0.                                                        PD14
              GOO=0.                                                        PD14
              GOM=0.                                                        PD14
           10 POV=0.                                                        PDY3B
              POG=0.                                                        PDY3B
              POR=0.                                                        PDY3B
              POM=0.                                                        PDY3B
              POP=0.                                                        PDY3B
              POO=0.                                                        PDY3B
           20 CONTINUE                                                      PDY3B
              ROV=  SINGAM                                                  PDY3B
              ROG=  V*COSGAM                                                PDY3B
              OOV=  COSGAM* COSPSI/R                                        PDY3B
              OOG=  -V*SINGAM*COSPSI/R                                      PDY3B
              OOP=-V*COSGAM*SINPSI/R                                        PDY3B
              OOR=-V*OOV/R                                                  PD14
              RCLAM = R*COSRHO                                              PDY3B
              UOV=  COSGAM* SINPSI/RCLAM                                    PDY3B
              UOG= -V*SINGAM* SINPSI/ RCLAM                                 PDY3B
              UOP=  V*COSGAM*COSPSI/ RCLAM                                  PDY3B
              UOR=  -V* UOV/R                                               PDY3B
              UOO= -UOR*SINRHO*R/ COSRHO                                    PDY3B
              MOM=AMM                                                       PD14
              MOR=AMR                                                       PD14
              MOV=AMV                                                       PD14
              IF(QMULT*RO.NE.0.) GO TO 30                                   PDY3B  30
              HTOV=0.                                                       PDY3B
              HTOR=0.                                                       PDY3B
              GO TO 40                                                      PDY3B  40
           30 CONTINUE                                                      PDY3B
              HTO=  QMULT* 17600.*SQRT(RO/RHOB)*(V/26000.)**3.15            PDY3B
              HTOV= 3.15 *HTO/V                                             PDY3B
              HTOR= .5 * HTO*ROR/ RO .                                      PDY3B
           40 CONTINUE                                                      PDY3B
      C                                                                     COMN
      C         V       ZERO  BANK  ANGLE  CONSIDERATION                    COMN
           50 IF(JGII.NE.12) RETURN                                         PDY3B
              GOPH=0.                                                       PDY3B
              POPH=0.                                                       PDY3B
              RETURN                                                        PDY3B
              END                                                           PDY3B
```

20 OCT 72 6.01-46

SUBRØUTINE PRMSET

Subroutine PRMSET

Purpose

PRMSET sets up nominal values of adjustable parameters for next iteration or solution trajectory.

Description

PRMSET is called from TEST.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ICOR		I	Phase sequence array	/XCODES/(	10)	ADJUST	I	ICOR
						FNTG	I	ICOR
						PRMSET	I	ICOR
						SDINP	M	ICOR
IPOINT		I	Code for each adjustable parameter in steepest descent.	/PARAM /(	1)	ADJUST	I	IPOINT
						PRMSET	I	IPOINT
						SDINP	O	IPOINT
						STAU	I	IPOINT
						TOPM	D	IPOINT
NPARA		I	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	D	NPARA
NPH		I	Number of phases in trajectory	/XCODES/(	164)	BNTG	I	NPH
						FNTG	O	NPH
						PRMSET	I	NPH
						SDINP	M	NPH
						TEST	I	NPH
						TOPM	I	NPH
OMG	Ω_j	M	Array of arc cut off values [sd]	/GENF /(	1)	ADJUST	M	OMG
						FNTG	I	OMG
						PRMSET	M	OMG
						PROPB	I	OMG
						SDINP	M	OMG
						STP3	I	OMG
						TOPM	D	IOMG
OMGP		O	Array of phase cut off values [sd]	/GENF /(	21)	ADJUST	O	OMGP
						FNTG	M	OMGP
						PRMSET	O	OMGP
						SDINP	M	OMGP
PARA	p	M	Adjustable parameter nominal values.	/PARAM /(	252)	ADJUST	I	PARA
						PRMSET	M	PARA
						TOPM	D	PARA
SVAR	$y\vert_{t=0}$	M	Array of state values at initial problem time [sd]	/GENF /(	79)	ADJUST	O	SVAR
						BNTG	I	SVAR
						FNTG	I	SVAR
						PRMSET	M	SVAR
						REU3	I	SVAR
						SDINP	M	SVAR
						TEST	I	SVAR
						TOPM	I	SVAR
						TRTOSZ	I	SVAR

```
      SUBROUTINE PRMSET(II)                                              PRMSET
C                                                                        COMN
C           SETS  UP  NOMINAL VALUES OF  PARAMETERS FOR  NEXT            COMN
C           ITERATION                                                    COMN
C           DEPENDS  ON  WHETHER   LAST  TRAIL  TRAJECTORY               COMN
C           SERVES AS NEW  NOMINAL( II=0)  OR   PREVIOUS                 COMN
C           NOMINAL  WILL  BE  SOLUTION  TRAJECTORY (II GT G )           COMN
      COMMON/STATE3/                                                     STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)      ,SVY(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM     ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI     ,SINRHO    ,COSRHO    ,OCORHO     ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2    ,                                  STATE3D
     *VDV        ,GDV        ,RDV       ,MDV       ,PDV        ,ODV      , STATE3D
     *UDV        ,VDG        ,GDG       ,RDG       ,PDG        ,ODG      , STATE3D
     *UDG        ,VDR        ,GDR       ,MDR       ,PDR        ,ODR      , STATE3D
     *UDR        ,VDM        ,GDM       ,MDM       ,PDM        ,VDP      , STATE3D
     *GDP        ,PDP        ,ODP       ,UDP       ,VDO        ,GDO      , STATE3D
     *PDO        ,UDO        ,HTDV      ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                                STATE3D
      COMMON/STATE3/                                                     STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                     STATE3D
      COMMON/GENF/                                                       GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)   ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)   ,COTI(9,9) ,DCON(9)    ,DTP      , GENF
     *DTS        ,DY         ,G         ,DPSQ      ,Q          ,QS       , GENF
     *R          ,RE         ,MACH      ,PA        ,RO         ,CS       , GENF
     *VNU        ,PAR        ,ADR       ,CSR       ,VNR        ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH     ,TIMES     ,TOP       ,TOS        ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T          ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME       ,OMP      , GENF
     *TIMPR      ,LIFT       ,DRAG      ,TAX       ,TBURN      ,TBU(20)  , GENF
     *AE         ,FP         ,FPOLD     ,FPD       ,MACHR      ,MACHV    , GENF
     *QR         ,QV         ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA      ,           DRAGV     ,DRAGR      ,DRAGA    , GENF
     *            LIFTM      ,DBR       ,DB        ,ISP        ,ISPF     , GENF
     *                        ULFT      ,ULFTV     ,ULFTR      ,ULFTA    , GENF
     *XMCG       ,XMCGV      ,XMCGR     ,XMCGA     ,XMCGM      ,CODAE    , GENF
     *CULFT      ,CT         ,CALPHA    ,COE       ,DELTAE     ,SID      , GENF
     *COD        ,SIDAE      ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                   GENF
     *XJV        ,XJR        ,GH        ,GAMMAD    ,XKG        ,XKP      , GENF
     *FRATED     ,IRATED     ,                                            GENF
     *P1         ,P2         ,P3        ,XK1       ,XK2        ,XK3      , GENF
     *XK1T       ,XK2T       ,XK3T      ,XK1D      ,XK2D       ,XK3D     , GENF
     *XK1A       ,XK2A       ,XK3A      ,XK1V      ,XK2V       ,XK3V     , GENF
     *XK1G       ,XK2G       ,XK3G      ,XK1P      ,XK2P       ,XK3P     , GENF
     *XK1R       ,XK2R       ,XK3R      ,XK1Q      ,XK2Q       ,XK3Q     , GENF
     *XK1U       ,XK2U       ,XK3U      ,XK1M      ,XK2M       ,XK3M     , GENF
     *PV         ,PG         ,PP        ,PR        ,PQ         ,OPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                        GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                 GENF
      COMMON /XCODES/                                                    XCODES
     *ITQ    (9),ICOR  (20),ITI        ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)             ,NCNST     ,NSB        ,NSAB       ,NICNB    , XCODES
     *I20P       ,ICOP       ,IFAW      ,IFAR      ,IFB        ,IND      , XCODES
     *IOPEN      ,IPH        ,ISPH      ,ISST      ,IARC       ,ISTART   , XCODES
     *ITCT       ,ITER       ,IVAR      ,JK        ,JPS        ,JS       , XCODES
     *KOP        ,KPST       ,K         ,KST       ,NAD        ,NCASE    , XCODES
     *NCN        ,NEQB       ,NEQ       ,NOP       ,NPH        ,N        , XCODES
     *NST        ,IPST       ,IPRINT    ,ISTN      ,IFHN       ,ISTNB    , XCODES
     *IPHNB      ,IBLK1      ,IBLK2     ,ISTOP     ,ISTPP      ,L        , XCODES
     *IFOB       ,NB         ,LB        ,MB        ,NPHP       ,NPHB     , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3           XCODES
      COMMON/PARAM/                                                      PARAM
     *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),            PARAM
     *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                        PARAM
     *,DELP(9)                                                           PARAM
C         THIS  ROUTINE  SETS  UP  NOMINAL PARAMETER  VALUE              PRMSET
C         I      TEST  FOR   NORMAL  OR  ERROR  TYPE  CONVERGANCE STOP   PRMSET
      IF(II GT.0)  GO TO 30                                              PRMSET  30
C       IA  NORMAL   CONVERGANCE   SET                                   PRMSET
      DO 20 I=1,NPARA                                                    PRMSET
      IF(IPOINT(I) GT.21) GO TO 10                                       PRMSET  10
```

20 OCT 72 G 01-46

```
      KK= IPOINT(I)                                        PRMSET
C                                                          COMN
C                         ARC  TIME                        COMN
      PARA(I)= OMG(KK)                                     PRMSET
      GO TO 20                                             PRMSET
   10 KK = IPOINT(I)- 21                                   PRMSET
C                                                          COMN
C                          INITIAL   STATE                 COMN
      PARA(I) = SVAR(KK+1)                                 PRMSET
   20 CONTINUE                                             PRMSET
      RETURN                                               PRMSET
C       IB   ERROR IN CONVERGNCE                           PRMSET
   30 DO  60  I=1,NPARA                                    PRMSET
      IF(IPOINT(I).GT.21) GO TO 50                         PRMSET
      KK= IPOINT(I)                                        PRMSET
      OMG(KK)= PARA(I)                                     PRMSET
      DO 40 JJ=1,NPH                                       PRMSET
      IF(ICOR(JJ).NE.KK) GO TO 40                          PRMSET
      OMGP(JJ,2)=PARA(I)                                   PRMSET
   40 CONTINUE                                             PRMSET
      GO TO  60                                            PRMSET
   50 KK= IPOINT(I)-21                                     PRMSET
      SVAR(KK+1) = PARA(I)                                 PRMSET
   60 CONTINUE                                             PRMSET
      RETURN                                               PRMSET
      END                                                  PRMSET
```

20 OCT 72 G 01-46

SUBRØUTINE
PRØPB

Subroutine PROPB

Purpose

Subroutine performs arc initialization for adjoint solution.

Description

PRØPB is called from BNTG.

593

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
AE	A_{exit}	O	Total nozzle exit area		/GENF /(	520)	ACCEL	I	AE
							FH2	I	AE
							IMPUL	I	AE
							PROPB	O	AE
							PROPIN	O	AE
							SDER3	I	AE
AEZRO	α_{old}	O	Angle of attack from last nominal trajectory	(DEG)	/AEC03 /(	1)	AST3	M	APHO
							FNTG	I	APHO
							MTX3A	I	APHO
							OUT	I	APHO
							PROPB	O	AEZRO
							PROPIN	O	AEZRO
CDE		O	Constant value of engine deflection	(RAD)	/GENF /(	553)	EL1	I	CDE
							PROPB	O	CDE
							PROPIN	O	CDE
DRAG	D	O	Aerodynamic drag	(LBS)	/GENF /(	497)	ACCEL	I	DRAG
							BL5	I	DRAG
							BL7	I	DRAG
							BL8	I	DRAG
							ENVPRM	I	DRAG
							FH3	I	DRAG
							OUT	I	DRAG
							PROPB	O	DRAG
							PROPIN	O	DRAG
							SDER3	I	DRAG
							VT	M	DRAG
EJ	A_{exit}	I	Nozzle exit area	(FT^2)	/ARCDAT/(	2)	PROPB	I	EJ
							PROPIN	I	EJ
							SIZIN	O	EJ
FRATE		I	Input rated vacuum thrust per engine	(LBS)	/ARCDAT/(	42)	EQUA3	I	FRATE
							FXDAT	I	FRATE
							PROPB	I	FRATE
							PROPIN	I	FRATE
							SIZIN	O	FRATE
FRATED		O	Net rated maximum rocket vacuum thrust	(LBS)	/GENF /(	567)	IMPUL	I	FRATED
							PROPB	O	FRATED
							PROPIN	O	FRATED
FVAC		M	Total vacuum thrust [rocket]	(LBS)	/GENF /(	528)	ACCEL	I	FVAC
							EQUA3	M	FVAC
							FH2	I	FVAC
							IMPUL	M	FVAC
							PROPB	M	FVAC
							PROPIN	M	FVAC
							SDER3	I	FVAC
GAMMAD		O	Pitch rate	(RAD)	/GENF /(	564)	BL4	I	GAMMAD
							PROPB	O	GAMMAD
							PROPIN	O	GAMMAD
GMAX	G_{MAX}	I	Maximum total acceleration g load		/ARCDAT/(	12)	BL5	I	GMAX
							FH3	I	GMAX
							MODELA	I	GMAX
							PROPB	I	GMAX
							PROPIN	I	GMAX
GMDOT	$\dot{\gamma}*$	I	Pitch rate	(DEG/SEC)	/ARCDAT/(	15)	DER3A	I	GMDOT
							MODELA	I	GMDOT
							MODELB	I	GMDOT
							PROPB	I	GMDOT
							PROPIN	I	GMDOT
HDMAX	$\dot{Q}_{MAX}$	I	Maximum heating rate inequality constraint		/ARCDAT/(	14)	PROPB	I	HDMAX

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IARC		I	Arc number		/XCODES/(	146)	ADICB3	I	IARC
							ADID3A	I	IARC
							ADJUST	I	IARC
							AST3	I	IARC
							BNTG	M	IARC
							ENVPRM	I	IARC
							FNTG	M	IARC
							GETIT	I	IARC
							MODELA	I	IARC
							PROPB	I	IARC
							PROPIN	I	IARC
							REU3	I	IARC
							SDINP	M	IARC
							STAU	I	IARC
							STP3	I	IARC
							TRTOSZ	I	IARC
INEQFL		I	A 20 word array that contains the code number of the state variables inequality constraint that applies on each subarc A zero entry indicates that no SVIC applies.		/GLOBAL/(	73)	PROPB	I	INEQFL
							PROPIN	M	INEQFL
							STP3	M	INEQFL
JAER		I	Aerodynamic model option flag		/ARCDAT/(	9)	BEROCO	I	JAER
							EQUA3	I	JAER
							GEINP	I	JAER
							OUT	I	JAER
							PROPB	I	JAER
							PROPIN	I	JAER
							VT	I	JAER
JPRO		I	Propulsion model option flag		/ARCDAT/(	10)	EQUA3	I	JPRO
							GEINP	I	JPRO
							IMPUL	I	JPRO
							MODELA	I	JPRO
							PROPB	I	JPRO
							PROPIN	I	JPRO
JPRP		O	Propulsion flag for different rocket options		/XCODES/(	194)	ACCEL	I	JPRP
							DER3A	I	JPRP
							EQUA3	I	JPRP
							MODELA	I	JPRP
							PDY3A	I	JPRP
							PROPB	O	JPRP
							PROPIN	O	JPRP
JP1		O	Option flag for first governing equation		/XCODES/(	217)	AGETB3	M	JP1
							AST3	M	JP1
							MODELA	M	JP1
							MODELB	I	JP1
							PROPB	O	JP1
							PROPIN	O	JP1
JP2		O	Option flag for second governing equation		/XCODES/(	218)	MODELA	I	JP2
							MODELB	I	JP2
							PROPB	O	JP2
							PROPIN	O	JP2
JS		I	Absolute value of arc cut-off option code		/XCODES/(	153)	ADICB3	M	JS
							ADIC3A	I	JS
							ADID3A	I	JS
							BNTG	M	JS
							FNTG	M	JS
							PROPB	I	JS
							PROPIN	I	JS
							STP3	I	JS
							TOL3	I	JS
LIFT	L	O	Aerodynamic lift	(LBS)	/GENF /(	496)	ACCEL	I	LIFT
							BL4	I	LIFT
							BL5	I	LIFT
							BL6	I	LIFT
							ENVPRM	I	LIFT
							FH3	I	LIFT
							OUT	I	LIFT
							PROPB	O	LIFT
							PROPIN	O	LIFT
							VT	O	LIFT

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
MAEA		I	Curve number	/ARCDAT/(	18)	EQUA3	I	MAEA
						GEINP	I	MAEA
						PROPB	I	MAEA
						PROPIN	I	MAEA
MISP		I	Curve number kISP loss table	/ARCDAT/(	26)	IMPUL	I	MISP
						PROPB	I	MISP
						PROPIN	I	MISP
MPIN		I	Save thrust curve numbers for adjoint solution	/XCODES/(	197)	PROPB	I	MPIN
						PROPIN	O	MPIN
MT		I	Curve number -thrust table	/ARCDAT/(	25)	FXDAT	I	MT
						PROPB	I	MT
						PROPIN	I	MT
						THRUST	I	MT
MTT		M	Thrust curve number	/XCODES/(	196)	EQUA3	I	MTT
						PROPB	M	MTT
						PROPIN	M	MTT
NST		I	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST
OMG	Ω_j	I	Array of arc cut off values [sd]	/GENF /(	1)	ADJUST	M	OMG
						FNTG	I	OMG
						PRMSET	M	OMG
						PROPB	I	OMG
						SDINP	M	OMG
						STP3	I	OMG
						TOPM	O	IOMG
QMAX	q_{MAX}	I	Maximum dynamic pressure instantaneous inequality limit (PSF)	/ARCDAT/(	11)	PROPB	I	QMAX
RDI		I	Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	O	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI
REMAX	$R_{ey_{MAX}}$	I	Maximum unit reynolds number inequality constraint	/ARCDAT/(	41)	PROPB	I	REMAX
T	T	O	Thrust (LBS)	/GENF /(	411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SDER3	I	T
TBU		I	Saved rocket burn initiation times used during adjoint integration (sd)	/GENF /(	500)	PROPB	I	TBU
						PROPIN	O	TBU

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
TBURN	t_b	O	Rocket burn initiation time on forward trajectory[sd]	/GENF /(	499)	EQUA3 MODELA PROPB PROPIN	I I O M	TBURN TBURN TBURN TBURN
TMULT	T_{mult}	I	Thrust multiplier or number of engines	/ARCDAT/(	4)	EQUA3 FXDAT PROPB PROPIN SIZIN	I M I I O	TMULT TMULT TMULT TMULT TMULT
ZERO		O	Partial of mach number wrt altitude	/GENF /(	524)	EQUA3 PROPB PROPIN VT	O O O I	MACHR ZERO ZERO MACHR

30 OCT 72 G 01-46

```
      SUBROUTINE  PROPB                                                  PROPB
C                                                                        COMN
C        PERFORMS  ARC  INITIALIZATION FOR ADJOINT  SOLUTION             COMN
C                                                                        COMN
      COMMON/AEC03/                                                      AEC03
     *APHO     ,APHR     ,ALPHA    ,VDA      ,GDA      ,PDA        ,     AEC03
     *SINA     ,COSA     ,PHIO     ,PHID     ,PHI      ,SINPHI     ,     AEC03
     *COSPHI   ,GDPH     ,PDPH     ,XLAMA(9) ,XLAMP(9) ,CDG        ,     AEC03
     *CDOM     ,CLO      ,FK       ,XCGM     ,ZCGM     ,CLOM       ,     AEC03
     *CM       ,CMA      ,CMAM ,CMM ,CMD     ,CMGM     ,FKM        ,     AEC03
     *CLAM     ,CL       ,CLA      ,CLM      ,                           AEC03
     *CD       ,CDA      ,CDM                                            AEC03
      EQUIVALENCE (APHO,AEZRO)                                           APR27
      DIMENSION AEZRO(1)                                                 APR27
      COMMON /XCODES/                                                    XCODES
     *ITQ    (9),ICOR  (20),ITI     ,INTB     ,JGIO(20,2),JPH (20,2),    XCODES
     *JST   (20)        ,NCNST    ,NSB      ,NSAB     ,NICNB      ,     XCODES
     *I2OP     ,ICOP     ,IFAW     ,IFAR     ,IFB      ,IND        ,     XCODES
     *IOPEN    ,IPH      ,ISPH     ,ISST     ,IARC     ,ISTART     ,     XCODES
     *ITCT     ,ITER     ,IVAR     ,JK       ,JPS      ,JS         ,     XCODES
     *KOP      ,KPST     ,K        ,KST      ,NAD      ,NCASE      ,     XCODES
     *NCN      ,NEQB     ,NEQ      ,NOP      ,NPH      ,N          ,     XCODES
     *NST      ,IPST     ,IPRINT   ,ISTN     ,IPHN     ,ISTNB      ,     XCODES
     *IPHNB    ,IBLK1    ,IBLK2    ,ISTOP    ,ISTPP    ,L          ,     XCODES
     *IFOB     ,NB       ,LB       ,MB       ,NPHP     ,NPHB       ,     XCODES
     *NCTIN    ,NEQF ,ILAB(8),JFRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON/GENF/                                                       GENF
     *OMG(20)  ,OMGP(20,2),VARQ(9) ,TOL(9)   ,SVAR(10) ,WDC(20)    ,     GENF
     *A(9,9)   ,ACON(9)  ,BCON(9)  ,COTI(9,9) ,DCON(9) ,DTP        ,     GENF
     *DTS      ,DT       ,G        ,DPSQ     ,Q        ,QS         ,     GENF
     *R        ,RE       ,MACH     ,PA       ,RO       ,CS         ,     GENF
     *VNU      ,PAR      ,ROR      ,CSR      ,VNR      ,SUMSQ      ,     GENF
     *SVSQ     ,TIMEPH   ,TIMES    ,TOP      ,TOS      ,TR(9)      ,     GENF
     *TST(20)  ,TPH   (20),DIS(20) ,DIP(20)  ,T        ,W          ,     GENF
     *TLP1(20) ,TLS1  (20),DIP1(20) ,DIS1(20) ,TIME    ,OMP        ,     GENF
     *TIMPR    ,LIFT     ,DRAG     ,TAX      ,TBURN    ,TBU(20)    ,     GENF
     *AE       ,FP       ,FPOLD    ,FPD      ,MACHR    ,MACHV      ,     GENF
     *QR       ,QV       ,FVAC     ,LIFTV    ,                             GENF
     *LIFTR    ,LIFTA    ,          DRAGV    ,DRAGR    ,DRAGA      ,     GENF
     *          LIFTM    ,DBR      ,DB       ,ISP      ,ISPF       ,     GENF
     *                    ULFT     ,ULFTV    ,ULFTR    ,ULFTA      ,     GENF
     *XMCG     ,XMCGV    ,XFCGR    ,XMCGA    ,XMCGM    ,CODAE      ,     GENF
     *CULFT    ,CT       ,CALPHA   ,CDE      ,DELTAE   ,SID        ,     GENF
     *CDO      ,SIDAE    ,XCG      ,ZCG      ,XJ                         GENF
      COMMON / GENF  /                                                   GENF
     *XJV      ,XJR      ,GH       ,GAMMAD   ,XKG      ,XKP        ,     GENF
     *FRATED   ,IRATED   ,                                               GENF
     *P1       ,P2       ,P3      ,XK1       ,XK2      ,XK3        ,     GENF
     *XK1T     ,XK2T     ,XK3T     ,XK1D     ,XK2D     ,XK3D       ,     GENF
     *XK1A     ,XK2A     ,XK3A     ,XK1V     ,XK2V     ,XK3V       ,     GENF
     *XK1G     ,XK2G     ,XK3G     ,XK1P     ,XK2P     ,XK3P       ,     GENF
     *XK1R     ,XK2R     ,XK3R     ,XK1O     ,XK2O     ,XK3O       ,     GENF
     *XK1U     ,XK2U     ,XK3U     ,XK1M     ,XK2M     ,XK3M       ,     GENF
     *PV       ,PG       ,PP       ,PR       ,PO       ,DPDY(3,8)        GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,       GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                   FRAT
      DIMENSION TPH1(10),TST1(10)                                        GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                 GENF
      COMMON/ARCDAT/                                                     ARCDAT
     *SREF     ,EJ       ,XISP     ,TMULT    ,DTNC     ,DTPI       ,     ARCDAT
     *IATM     ,IMODE    ,JAER     ,JPRO     ,QMAX     ,GMAX       ,     ARCDAT
     *XLMAX    ,HDMAX    ,GMDOT    ,ALFMAX   ,PHMAX    ,MAEA       ,     ARCDAT
     *MAEB     ,MAEC     ,MAED     ,MAEE     ,MAEF     ,MAEG       ,     ARCDAT
     *MT       ,MISP     ,MXCG     ,MZCG     ,MWDA     ,MWDB       ,     ARCDAT
     *MDB      ,XCGR     ,ZCGR     ,XE       ,ZE       ,XT         ,     ARCDAT
     *DREF     ,MCND     ,RHOB     ,QMULT    ,REMAX                      ARCDAT
     * ,FRATE  , ARCD(9)                                                 RETAP
      DIMENSION  ARCDA(40)                                               ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                            ARCDAT
      COMMON/DATA/                                                       DATA
     *PI       ,RAD      ,RDI      ,SC       ,UMF      ,TMPF       ,     DATA
     *FTNM     ,CAR      ,JOP1     ,JOP2     ,JOP3     ,JOP4       ,     DATA
      DIMENSION ZERO(110)                                                P014
      EQUIVALENCE (ZERO,MACHR)                                           P014
      COMMON/GLOBAL/                                                     GLOBAL
```

```
      *GR        ,ER        ,OMGZ      ,XLAMRF     ,YMURF     ,LUM          GLOBAL
      *,JJOP(10) ,IFATAL    ,NARC       ,NBRAN     ,NFARC     ,ID(4)        GLOBAL
      *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                                  GLOBAL
      *,GM       ,PSIRF,IPFLG1,       IPFLG2,IPFLG3,IPFLG4,INEQFL(20)       GLOBAL
      *,ITPSO  ,KSOL     ,KGLOBL(8)                                         RETAP
       EQUIVALENCE(INQF,INEQFL(20))                                         NOS
       EQUIVALENCE(INEDS,ILAB(2))                                           APR27
C                                                                           CORN
C              ZERO  COMPUTATION AREAS  IN  COMMON                          COMN
        DO 1 I=1,110                                                        PD14
     1 ZERO(I)=0.                                                           PD14
       DO 2 I=1,54                                                          APR27
     2 AEZRO(I)=0.                                                          APR27
       LIFT=0.                                                              APR27
       DRAG=0.                                                              APR27
       GAMMAD=GMDOT*RDI                                                     PD14
C      I    INITIALIZ BIVARIATE AERO                                        PROPB
       IF(IARC.EQ.NST) JJJ=999                                              PROPB
       IF(JAER.NE.2) GO TO 30                                               PROPB
       IF(JJJ.NE.999) GO TO   20                                            PROPB
    10 JJJ=MAEA                                                             PROPB
       CALL INBVAD(JJJ)                                                     PROPB
       GO TO 30                                                             PROPB
    20 IF(JJJ.NE.MAEA) GO TO 10                                             PROPB
    30 CONTINUE                                                             PROPB
C             INITIALIZE AIRBREATHER                                        FIXED
       IF(IARC.EQ.NST) KKK =999                                             FIXED
       IF(JPRO.NE.2)  GO  TO  350                                           FIXED
       IF(KKK.NE.999) GO TO 320                                             FIXED
   310 KKK = MWDB                                                           FIXED
       CALL  INBVPD(KKK)                                                    FIXED
       GO TO 350                                                            FIXED
   320 IF (KKK.NE.MWDB) GO TO 310                                           FIXED
   350 CONTINUE                                                             FIXED
C       I-A SET UP FLAG FOR INEQUALITY CUTT OFF (ADJOIMT DISCONTINUITY)     NOS
       INEDS=0                                                              APR27
       IF(IARC.LE.1) GO TO 39                                               NOS
       IF(INEQFL(IARC-1)) 31,31,39                                          NOS
    31 IF(JS-33) 39,32,33                                                   NOS
    32 IF(QMAX.EQ.OMG(IARC-1)) INEDS=1                                      APR27
       GO TO 39                                                             NOS
    33 IF(JS-35) 34,35,39                                                   NOS
    34 IF(HDMAX.EQ.OMG(IARC-1)) INEDS=1                                     APR27
       GO TO 39                                                             NOS
    35 IF(REMAX.EQ OMG(IARC-1)) INEDS=1                                     APR27
    39 CONTINUE                                                             NOS
C      II 'TEST FOR  MOMENT BALANCE AND INITIALIZE  ENG. DEFL.              PROPB
       JP2 = 1                                                              PROPB
       CDE = 0.                                                             PROPB
       IF(JAER.NE.3) GO TO 40                                               PROPB
       JP2= 2                                                               PROPB
    40 CONTINUE                                                             PROPB
C    II-A  TEST  FOR  AIRBREATHER                                           FIXED
       IF(JPRO.NE.2)  GO  TO  45                                            FIXED
       JP1= 4                                                               FIXED
       JPRP=1                                                               FIXED
       RETURN                                                               FIXED
    45 CONTINUE                                                             FIXED
C       III  INITIALIZE  PROPULSION  BURN TIME  AND CURVE NUMBER            PROPB
       AE=EJ*TMULT                                                          PROPB
       IF(AT) 50,90,50                                                      PROPB
    50 TBURN = TBU(IARC)                                                    PROPB
       ATT   = MPIN(IARC)                                                   PROPB
C                                                                           CORN
C        III-A  TEST  FOR  ACCEL  LIMIT  AND SET  ROCKET FLAG               CORN
       IF(GMAX)60,60,70                                                     PROPB
    60 JPRP= 2                                                              PROPB
```


605

```
      GO TO  80                                                   PROPB
   70 JPRP =3                                                     PROPB
C                                                                 COMN
C     III-B       THROTTLING  AND DUAL ENGINE   LOGIC             COMN
   80 IF(MISP.EQ.0.AND.JPRO.EQ.0) RETURN                          AAA
      IF(JPRO.EQ.1) CALL ISPIN                                    AAA
      IF(FRATE.EQ.0 ) GO TO 85                                    FRAT
   82 FRATED = FRATE * TMULT                                      FRAT
      RETURN                                                      FRAT
   85 CALL  SPLYNE(MTT, 0., FRATE, DUM)                           FRAT
      GO TO 82                                                    APR72
   90 JPRP=1                                                      PROPB
      FVAC=0.                                                     PO14
      T=0.                                                        PO14
C                                                                 COMN
C      III-C      USE   RATED  THRUST IF  INPUT                   COMN
      IF(FRATE.LE.0.) RETURN                                      FRAT
      AE = EJ*TMULT                                               FRAT
      MTT=0                                                       PH1SZ
      FVAC = FRATE * TMULT                                        FRAT
      IF(JPRO.EQ.1) CALL ISPIN                                    AAA
      FRATED = FVAC                                               FRAT
      IF(GMAX.GT.0.) GO TO 100                                    FRAT
      JPRP =2                                                     FRAT
      RETURN                                                      FRAT
  100 JPRP = 3                                                    FRAT
      T=FVAC                                                      PH1SZ
      RETURN                                                      APR
      END                                                         PROPB
```

609

SUBRØUTINE PRØPIN

Subroutine PROPIN

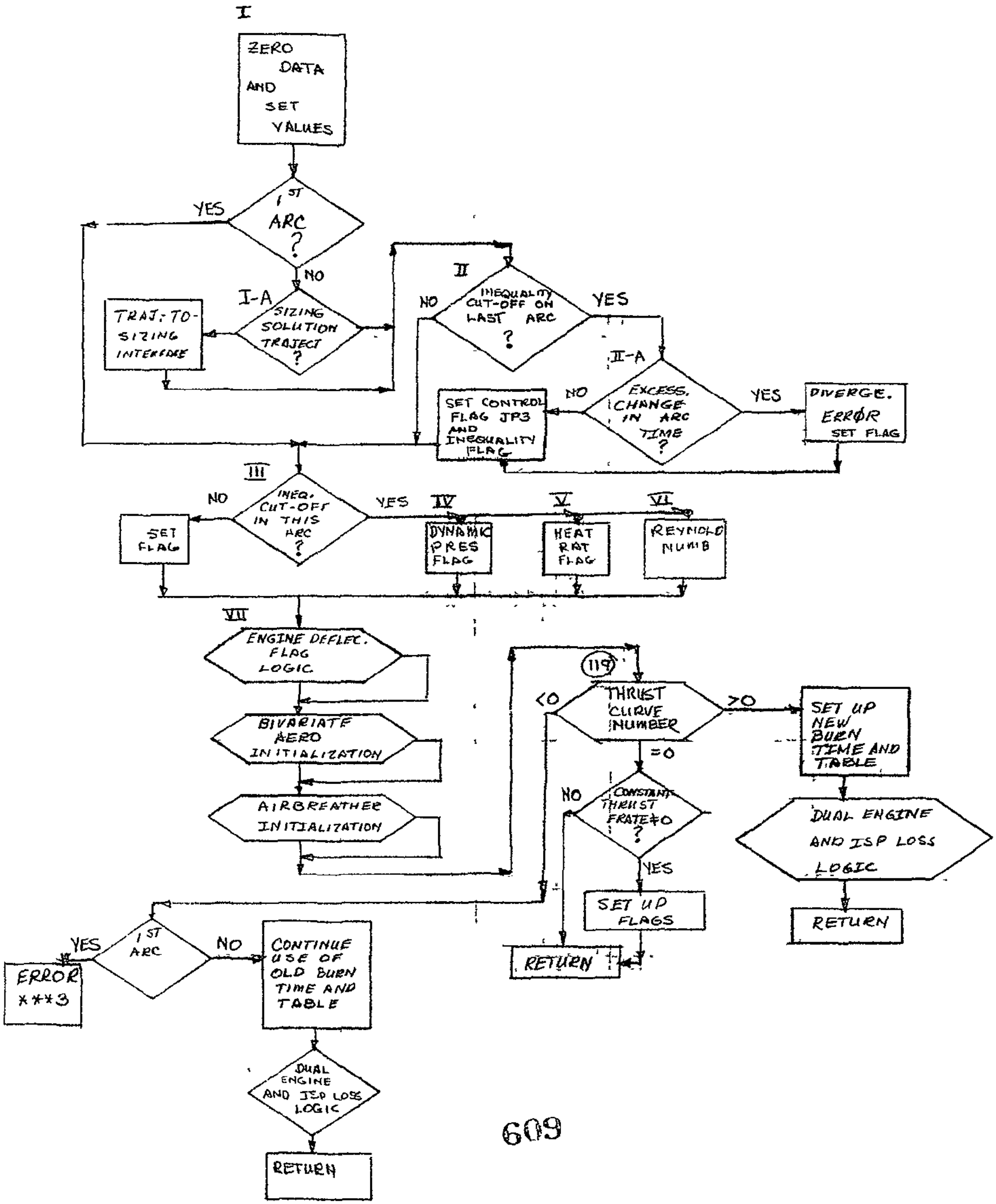
Purpose

Subroutine PROPIN performs arc initialization for the forward trajectory.

Description

PROPIN is called from FNTG.

SUBROUTINE PROPIN



663

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AE	A_{exit}	O	Total nozzle exit area		/GENF /(	520)	ACCEL	I	AE
							FH2	I	AE
							IMPUL	I	AE
							PROPB	O	AE
							PROPIN	O	AE
							SDER3	I	AE
AEZRO	α_{old}	O	Angle of attack from last nominal trajectory	(DEG)	/AEC03 /(	1)	AST3	M	APHO
							FNTG	I	APHO
							MTX3A	I	APHO
							OUT	I	APHO
							PROPB	O	AEZRO
							PROPIN	O	AEZRO
CDE		O	Constant value of engine deflection	(RAD)	/GENF /(	553)	EL1	I	CDE
							PROPB	O	CDE
							PROPIN	O	CDE
DRAG	D	O	Aerodynamic drag	(LBS)	/GENF /(	497)	ACCEL	I	DRAG
							BL5	I	DRAG
							BL7	I	DRAG
							BL8	I	DRAG
							ENVPRM	I	DRAG
							FH3	I	DRAG
							OUT	I	DRAG
							PROPB	O	DRAG
							PROPIN	O	DRAG
							SDER3	I	DRAG
							VT	M	DRAG
DTNC	$\Delta\tau$	I	Integration interval	(SEC)	/ARCDAT/(	5)	BNTG	I	DTNC
							FNTG	I	DTNC
							GEINP	M	DTNC
							PROPIN	I	DTNC
DVAR	$\dot{y}$	O	State vector derivatives in steepest descent module		/STATE3/(	15)	ADICB3	M	DVAR
							ADIC3A	I	DVAR
							ADID3A	M	DVAR
							DER3A	O	VD
							DTF3	I	VT
							ENVPRM	I	DVAR
							POBC	I	VD
							PROPIN	O	DVAR
							REU3	I	DVAR
							RKTA3A	I	DY
							SDER3	O	DVAR
							STP3	I	DVAR
							YREF3	I	DVAR
							YREF3	I	VT
EJ	A_{exit}	I	Nozzle exit area	(FT^2)	/ARCDAT/(	2)	PROPB	I	EJ
							PROPIN	I	EJ
							SIZIN	O	EJ
FRATE		I	Input rated vacuum thrust per engine	(LBS)	/ARCDAT/(	42)	EQUA3	I	FRATE
							FXDAT	I	FRATE
							PROPB	I	FRATE
							PROPIN	I	FRATE
							SIZIN	O	FRATE
FRATED		O	Net rated maximum rocket vacuum thrust	(LBS)	/GENF /(	567)	IMPUL	I	FRATED
							PROPB	O	FRATED
							PROPIN	O	FRATED
FVAC		M	Total vacuum thrust [rocket]	(LBS)	/GENF /(	528)	ACCEL	I	FVAC
							EQUA3	M	FVAC
							FH2	I	FVAC
							IMPUL	M	FVAC
							PROPB	M	FVAC
							PROPIN	M	FVAC
							SDER3	I	FVAC
GAMMAD		O	Pitch rate	(RAD)	/GENF /(	564)	BL4	I	GAMMAD
							PROPB	O	GAMMAD
							PROPIN	O	GAMMAD

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
GMAX	G_{MAX}	I	Maximum total acceleration g load	/ARCDAT/(	12)	BL5	I	GMAX
						FH3	I	GMAX
						MODELA	I	GMAX
						PROPB	I	GMAX
						PROPIN	I	GMAX
GMDOT	$\dot{\gamma}^*$	I	Pitch rate (DEG/SEC)	/ARCDAT/(	15)	DER3A	I	GMDOT
						MODELA	I	GMDOT
						MODELB	I	GMDOT
						PROPB	I	GMDOT
						PROPIN	I	GMDOT
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
IND		I	Flag indicates whether on first nominal trajectory (IND=1)	/XCODES/(	141)	AST3	I	IND
						BGET3	I	IND
						FNTG	M	IND
						GUI3A	I	IND
						MTX3A	I	IND
						PROPIN	I	IND
INEQFL		M	A 20 word array that contains the code number of the state variables inequality constraint that applies on each subarc. A zero entry indicates that no SVIC applies.	/GLOBAL/(	73)	PROPB	I	INEQFL
						PROPIN	M	INEQFL
						STP3	M	INEQFL
IOPEN		O	Closed to open-loop control switch when equal to 2	/XCODES/(	142)	FNTG	M	IOPEN
						PROPIN	O	IOPEN
ISTART		O	Initialization and divergance flag	/XCODES/(	147)	AST3	O	ISTART
						BLGCON	O	ISTART
						BLYNE	O	ISTART
						FNTG	I	ISTART
						MODELA	O	ISTART
						PROPIN	O	ISTART
						REU3	I	ISTART
						TEST	M	ISTART
						TOPM	M	ISTART
ITER		I	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
						FNTG	I	ITER
						GETIT	I	ITER
						MODELA	I	ITER
						OUT	I	ITER
						PAYO2	M	ITER
						PROPIN	I	ITER
						TEST	M	ITER
						TOPM	M	ITER
JAER		I	Aerodynamic model option flag	/ARCDAT/(	9)	BEROCO	I	JAER
						EQUA3	I	JAER
						GEINP	I	JAER
						OUT	I	JAER
						PROPB	I	JAER
						PROPIN	I	JAER
						VT	I	JAER
JPRO		I	Propulsion model option flag	/ARCDAT/(	10)	EQUA3	I	JPRO
						GEINP	I	JPRO
						IMPUL	I	JPRO
						MODELA	I	JPRO
						PROPB	I	JPRO
						PROPIN	I	JPRO

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
JPRP		O	Propulsion flag for different rocket options	/XCODES/(	194)	ACCEL	I	JPRP
						DER3A	I	JPRP
						EQUA3	I	JPRP
						MODELA	I	JPRP
						POY3A	I	JPRP
						PROPB	O	JPRP
						PROPIN	O	JPRP
JP1		O	Option flag for first governing equation	/XCODES/(	217)	AGETB3	M	JP1
						AST3	M	JP1
						MODELA	M	JP1
						MODELB	I	JP1
						PROPB	O	JP1
						PROPIN	O	JP1
JP2		O	Option flag for second governing equation	/XCODES/(	218)	MODELA	I	JP2
						MODELB	I	JP2
						PROPB	O	JP2
						PROPIN	O	JP2
JP3		O	Option flag for third governing equation	/XCODES/(	219)	AGETB3	O	JP3
						AST3	M	JP3
						MODELA	M	JP3
						MODELB	I	JP3
						OUT	I	JP3
						PROPIN	O	JP3
JS		I	Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
						ADIC3A	I	JS
						ADID3A	I	JS
						BNTG	M	JS
						FNTG	M	JS
						PROPB	I	JS
						PROPIN	I	JS
						STP3	I	JS
						TOL3	I	JS
LIFT	L	O	Aerodynamic lift (LBS)	/GENF /(	496)	ACCEL	I	LIFT
						BL4	I	LIFT
						BL5	I	LIFT
						BL6	I	LIFT
						ENVPRM	I	LIFT
						FH3	I	LIFT
						OUT	I	LIFT
						PROPB	O	LIFT
						PROPIN	O	LIFT
						VT	O	LIFT
MAEA		I	Curve number	/ARCDAT/(	18)	EQUA3	I	MAEA
						GEINP	I	MAEA
						PROPB	I	MAEA
						PROPIN	I	MAEA
MISP		I	Curve number KISP loss table	/ARCDAT/(	26)	IMPUL	I	MISP
						PROPB	I	MISP
						PROPIN	I	MISP
MPIN		O	Save thrust curve numbers for adjoint solution	/XCODES/(	197)	PROPB	I	MPIN
						PROPIN	O	MPIN
MT		I	Curve number -thrust table	/ARCDAT/(	25)	FXDAT	I	MT
						PROPB	I	MT
						PROPIN	I	MT
						THRUST	I	MT
MTT		M	Thrust curve number	/XCODES/(	196)	EQUA3	I	MTT
						PROPB	M	MTT
						PROPIN	M	MTT
NARC	N_3	I	Number of subarcs in the problem.	/GLOBAL/(	18)	FNTG	I	NARC
						GEINP	M	NARC
						PROPIN	I	NARC
						SDINP	I	NARC
						SIZIN	I	NARC

30 OCT 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
RDI		I	Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO'	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	O	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI
T	T	O	Thrust (LBS)	/GENF /(	411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						ELZ	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SDER3	I	T
TBU		O	Saved rocket burn initiation times used during adjoint integration (sd)	/GENF /(	500)	PROPB	I	TBU
						PROPIN	O	TBU
TBURN	t_b	M	Rocket burn initiation time on forward trajectory[sd]	/GENF /(	499)	EQUA3	I	TBURN
						MODELA	I	TBURN
						PROPB	O	TBURN
						PROPIN	M	TBURN
TIME	t	I	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
TMULT	T_{mult}	I	Thrust multiplier or number of engines	/ARCDAT/(	4)	EQUA3	I	TMULT
						FXDAT	M	TMULT
						PROPB	I	TMULT
						PROPIN	I	TMULT
						SIZIN	O	TMULT
TST1		I	Arc end times for nominal trajectory	/GENF /(	433)	BNTG	I	TST1
						GETIT	I	TST1
						PROPIN	I	TST1
						SDINP	O	TST1
						TEST	O	TST1
						TOPM	I	TST1
						TRAN3	I	TST1
						TRTOSZ	I	TST1
ZERO		O	Partial of mach number wrt altitude	/GENF /(	524)	EQUA3	O	MACHR
						PROPB	O	ZERO
						PROPIN	O	ZERO
						VT	I	MACHR

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
ZZ		I	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A	I	VARL
						ADICB3	O	VARL
						ADIC3A	O	VARL
						PROPIN	I	ZZ
						RKTA3A	M	F
						RKTB3A	M	Y
						STVRL3	O	VARL
						TRAN3	M	VARL
UN06.		O	File of all output data	/ UN06 /($	)	BLICO	O	UN06.
						BNDRYC	O	.UN06.
						CRASH	O	.UN06.
						FRENCH	O	UN06.
						FXDAT	O	.UN06
						GEINP	O	.UN06.
						HUNT	O	.UN06.
						INEDIT	O	UN06
						ITERB	O	.UN06.
						MODELA	O	UN06
						MOMJ	O	.UN06.
						MPSI	O	.UN06.
						OUT	O	.UN06.
						PAYO2	O	.UN06.
						PRINT	O	UN06.
						PRINTV	O	UN06
						PRINTW	O	UN06.
						PRITEQ	O	.UN06.
						PRITVA	O	UN06.
						PROPIN	O	.UN06.
						PROTHR	O	.UN06.
						PRWTSM	O	.UN06.
						RANGE	O	.UN06.
						S	O	.UN06.
						SDINP	O	.UN06.
						SIZE	O	.UN06.
						SIZIN	O	.UN06.
						SIZOUT	O	.UN06.
						SOLVE	O	.UN06.
						SPLICO	O	.UN06.
						SPLIZ	O	.UN06.
						SPLYNE	O	UN06.
						SSSP	O	.UN06.
						STAU	O	.UN06.
						STPIT	O	.UN06.
						SUMOUT	O	.UN06.
						TABIN	O	.UN06.
						TEST	O	.UN06.
						VEHOF	O	.UN06.
						WTSCH	O	.UN06.
						WTVOL	O	UN06.

```
      SUBROUTINE  PROPIN                                                 PROPIN
C          THIS  ROUTINE  PERFORMS  ARC  INITIALIZATION                   PROPIN
C          FOR  FORWARD  TRAJECTORY                                      PROPIN
      COMMON/AECO3/                                                       AECO3
     *APHO       ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA        , AECO3
     *SINA       ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI     , AECO3
     *COSPHI     ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDG        , AECO3
     *CDGM       ,CLO        ,FK         ,XCGM       ,ZCGM       ,CLOM       , AECO3
     *CM         ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM       ,FKM        , AECO3
     *CLAM       ,CL         ,CLA        ,CLM        ,                         AECO3
     *CD         ,CDA        ,CDM                                              AECO3
      EQUIVALENCE (APHO,AEZRO)                                             APR27
      DIMENSION AEZRO(1)                                                   APR27
      COMMON/GENF/                                                         GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)     ,TOL(9)     ,SVAR(10)   ,WDC(20)    , GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)    ,COTI(9,9)  ,DCON(9)    ,DTP        , GENF
     *DTS        ,DT         ,G          ,DPSQ       ,Q          ,QS         , GENF
     *R          ,RE         ,MACH       ,PA         ,RO         ,CS         , GENF
     *VNU        ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ      , GENF
     *SVSQ       ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)      , GENF
     *TST(20)    ,TPH   (20),DIS(20)     ,DIP(20)    ,T          ,W          , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)    ,DIS1(20)   ,TIME       ,OMP        , GENF
     *TIMPR      ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)    , GENF
     *AE         ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV      , GENF
     *DR         ,QV         ,FVAC       ,LIFTV      ,                         GENF
     *LIFTR      ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA      , GENF
     *            LIFTM      ,DBR        ,DB         ,ISP        ,ISPF       , GENF
     *                        ULFT       ,ULFTV      ,ULFTR      ,ULFTA      , GENF
     *XMCG       ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE      , GENF
     *CULFT      ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SID        , GENF
     *COD        ,SIDAE      ,XCG        ,ZCG        ,XJ                       GENF
      COMMON / GENF  /                                                     GENF
     *XJV        ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP        , GENF
     *FRATED     ,IRATED     ,                                             GENF
     *P1         ,P2         ,P3         ,XK1        ,XK2        ,XK3        , GENF
     *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D       , GENF
     *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V       , GENF
     *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P       , GENF
     *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O       , GENF
     *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M       , GENF
     *PV         ,PG         ,PP         ,PR         ,PO         ,DPDY(3,8)    GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,        GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                      FRAT
      DIMENSION TPH1(10),TST1(10)                                          GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                   GENF
      COMMON /XCODES/                                                      XCODES
     *ITQ     (9),ICOR  (20),ITI        ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)             ,NCNST      ,NSB        ,NSAB       ,NICNB      , XCODES
     *I2OP       ,ICOP       ,IFAW       ,IFAR       ,IFB        ,IND        , XCODES
     *IOPEN      ,IPH        ,ISPH       ,ISST       ,IARC       ,ISTART     , XCODES
     *ITCT       ,ITER       ,IVAR       ,JK         ,JPS        ,JS         , XCODES
     *KOP        ,KPST       ,K          ,KST        ,NAD        ,NCASE      , XCODES
     *NCN        ,NEQB       ,NEQ        ,NOP        ,NPH        ,N          , XCODES
     *NST        ,IPST       ,IPRINT     ,ISTN       ,IPHN       ,ISTNB      , XCODES
     *IPHNB      ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP      ,L          , XCODES
     *IFOB       ,NB         ,LB         ,MB         ,NPHP       ,NPHB       , XCODES
     *NCTIN      ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3            XCODES
      COMMON/STATE3/                                                       STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99)  ,DVARL(99)  ,YO(9)      ,SVY(10)    , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15)  , STATE3D
     *SINPSI     ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2     , STATE3D
     *SVBV    (9),OMEGA      ,OMEGA2     ,                                  STATE3D
     *VDV        ,GDV        ,RDV        ,MDV        ,PDV        ,ODV        , STATE3D
     *UDV        ,VDG        ,GDG        ,RDG        ,PDG        ,ODG        , STATE3D
     *UDG        ,VDR        ,GDR        ,MDR        ,PDR        ,ODR        , STATE3D
     *UDR        ,VDM        ,GDM        ,MDM        ,PDM        ,VDP        , STATE3D
     *GDP        ,PDP        ,ODP        ,UDP        ,VDO        ,GDO        , STATE3D
     *PDO        ,UDO        ,HTDV       ,HTDR                                 STATE3D
      REAL MDM , MDV, MDR                                                  STATE3D
      COMMON/STATE3/                                                       STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                       STATE3D
      COMMON/GLOBAL/                                                       GLOBAL
     *GR         ,ER         ,OMGZ       ,XLAMRF     ,YMURF      ,LUM        GLOBAL
     *,JJOP(10)  ,IFATAL     ,NARC       ,NBRAN      ,NFARC      ,ID(4)      GLOBAL
     *,KTAB(20),ITAB(20)    ,SIG        ,MAXTAB                              GLOBAL
```

20 OCT 72 G 01-46

```
      *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)           GLOBAL
      *,ITPSO ,KSOL     ,KGLOBL(8)                                              RETAP
       REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                                 SIZING
       COMMON /SIZING/                                                         SIZING
C           PHASE II SIZING PARAMERERS                                         SIZING
      *TZ,      VV(3),     QP(14),   EROR,     PZ(5),     VQ,       SW(20), SIZING
      *SV(28),  SQ(37,5),  SE(11),   TLAT,     TLNG,                          SIZING
C          PHASE I SIZING PARAMERERS                                           SIZING
      *WBO,      WLOO,     DWEB,     DWEO,     TOLWT,     WPB,      TWRAT2, SIZING
      *BK1,      BK2,      BK3,      BK4,      ISIZE,     TRAFLG,   TWRATO, SIZING
      *OK1,      OK2,      OK3,      OK4,      PRFLG,     IPASS,    IPSMAX, SIZING
      *AEXIT,    TVACO,    NO,       WFO,      IDVEL,     ISPO,     ISPB,   SIZING
      *XPL,      TVACB,   NNB,       WEO,      WEB,       WO,       WLO,    SIZING
      *DVO,      DVB,      MUB,       MUO,      VSTG,      WPO              SIZING
      *,JTYP,    BECO ,    BSTG ,    ORBI,     ITNBW , ITNOW ,                SIZING
      * SVDPSQ  , SVDCON  ,IHUNT   ,IOPSTG  ,ISZD(19)                          UH
       COMMON/ARCDAT/                                                          ARCDAT
      *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI       ,  ARCDAT
      *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX       ,  ARCDAT
      *XLMAX     ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA       ,  ARCDAT
      *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG       ,  ARCDAT
      *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB       ,  ARCDAT
      *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT         ,  ARCDAT
      *DREF      ,MCND      ,RHOB      ,QMULT '   ,REMAX                    ARCDAT
      * ,FRATE  , ARCD(9)                                                     RETAP
       DIMENSION  ARCDA(40)                                                    ARCDAT
       EQUIVALENCE(SREF,ARCDA)                                                 ARCDAT
       DIMENSION ZERO(110),ZZ(41)                                              IO
       EQUIVALENCE( ZERO,MACHR),(ZZ,VARL),(ZZ(11),QMAX1),(ZZ(14),HDMAX1), PROPIN
      1(ZZ(41),REMAX1),(INDF,INEQFL(20))                                       PROPIN
       COMMON/DATA/                                                            DATA
      *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF       ,  DATA
      *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4          DATA
C       I   ZERO  OUT  COMPUTED  DATA IN  /SENF/  AND SET VALUES               PROPIN
       DO 10  I=1,110                                                          IO
    10 ZERO(I)= 0.                                                             PROPIN
       DO 2 I=1,54                                                             APR27
     2 AEZRO(I)=0.                                                             APR27
       LIFT=0.                                                                 APR27
       DRAG=0.                                                                 APR27
       GAMMAD=GMDOT*RDI                                                        PO14
       ILAB(1)=0                                                               PO14
       JP3 = 2                                                                 PROPIN
       DVAR(8)=0.                                                              APR27
       INQF =0                                                                 PROPIN
       IF(IARC.LE.1) GO TO 4                                                   PH1SZ   4
C      I-A  TEST FOR SIZING                                                    PH1SZ
       IF(ITER EQ.3.AND.JTYP.GT.0) CALL ARCEND                                 PH1SZ
     4 CONTINUE                                                                PH1SZ
C       II    TEST  AND  SET  INEQUALITY  FLAG   INQF                          PROPIN
       IF(IARC.EQ.1) GO TO  20                                                 PROPIN  20
       IF(INEQFL(IARC-1).EQ.0) GO TO 20                                        PROPIN  20
C       II-A TEST FOR EXCESSIVE CHANGE OF CUT-OFF TIME                         APR27
       IF(IND.EQ 1) GO TO 16                                                   PH1SZ   16
       TAT=ABS(TST1(IARC)-TIME)                                                APR27
       IF(TAT.LE.10.) GO TO 15                                                 APR27   15
       WRITE(6,12)                                                             APR27
    12 FORMAT (53H ****ERROR*** EXCESSIVE  CHANGE IN ARC CUT-OFF TIME    ) APR27
       ISTART=6                                                                APR27
    15 IF(TAT.GT 4.*DTNC) IOPEN=2                                              APR27
    16 CONTINUE                                                                PH1SZ
       INQF = INEQFL(IARC-1)                                                   PROPIN
       JP3 = INQF                                                              PROPIN
    20 IF(IARC.EQ.NARC) GO TO 60                                               PROPIN  60
C   III TEST FOR  WHETHER AN  INEQUALITY  CUTOFF  EXISTS  IN  CURRENT          PROPIN
C        ARC                                                                   PROPIN
       IF((JS-33)*(JS-34)*(JS-35)) 60,30,60                                    PROPIN  30  60
    30 CALL  READMS(9,ZZ,41,IARC+1)                                            PROPIN
C     IV    TEST  FOR   DYN. PRES. LIMIT '                                     PROPIN
       IF(QMAX1 EQ.0.) GO  TO  40                                              PROPIN  40
       IF(JS NE.33) GO TO  40                                                  PROPIN  40
       INEQFL(IARC) = 7                                                        PROPIN
       GO TO  70                                                               PROPIN  70
```

```
C     V      TEST    FOR  HEAT  RATE  LIMIT                                PROPIN
   40 IF(HDMAX1.EQ.0) GO TO  50                                            PROPIN
      IF(JS.NE.34) GO TO 50                                                PROPIN
      INEQFL(IARC) = 8                                                     PROPIN
      GO  TO  70                                                           PROPIN
C  VI  TEST   FOR   REYN. NO.  LIMIT                                       PROPIN
   50 IF(REMAX1.EQ.0.) GO  TO  60                                          PROPIN
      IF(JS.NE.35) GO TO 60                                                PROPIN
      INEQFL(IARC)= 9                                                      PROPIN
      GO  TO  70                                                           PROPIN
   60 INEQFL(IARC) = 0                                                     PROPIN
   70 CONTINUE                                                             PROPIN
C       VII   SET  BURN  TIME  AND  BIVARIATE  AREO  DATA                  PROPIN
C     ALSO TEST FOR DELTAE COMPUTATION                                     PROPIN
      JP2=2                                                                PROPIN
      IF(JAER EQ 3) GO TO 80                                             . PROPIN
      CDE=0.                                                               PROPIN
       JP2=1                                                               PROPIN
   80 CONTINUE                                                             PROPIN
      IF(IARC.EQ.1) JJJ=999                                                PROPIN
      IF(JAER.NE.2) GO TO 110                                              PROPIN
      IF(JJJ.NE.999) GO TO 100                                             PROPIN
   90 JJJ=MAEA                                                             PROPIN
      CALL INBVAD(JJJ)                                                     PROPIN
      GO TO 110                                                            PROPIN
  100 IF(MAEA.NE.JJJ) GO TO 90                                             FIXED
  110 CONTINUE                                                             PROPIN
C       INITIALIZE  AIRBREATHER                                            FIXED
      IF(IARC EQ 1) KKK= 999                                               FIXED
      IF(JPRO.NE.2) GO TO  119                                             FIXED
      IF(KKK.NE.999) GO TO 113                                             FIXED
  111 KKK= MWOB                                                            FIXED
      CALL INBVPD(KKK)                                                     FIXED
      GO TO 118                                                            FIXED
  113 IF(MWOB NE KKK) GO TO 111                                            FIXED
  118 JP1= 4                                                               FIXED
      JPRP= 1                                                              FIXED
      RETURN                                                               FIXED
  119 CONTINUE                                                             FIXED
      IF(MT)120,140,150                                                    PROPIN
  120 IF(IARC EQ 1)  GO TO 180                                             PROPIN
  130 TBU(IARC) = TBURN                                                    PROPIN
      MPIN(IARC)= MTT                                                      PROPIN
      IF(MISP.EQ.0) RETURN                                                 FRAT
      IF(MISP.EQ.0 AND.JPRO.EQ 0) RETURN                                   AAA
      IF(JPRO.EQ.1) CALL ISPIN                                             AAA
      IF(FRATE.EQ 0) GO TO 135                                             FRAT
  132 FRATED = FRATE * TMULT                                               FRAT
      RETURN                                                               FRAT
  135 CALL SPLYNE(MTT,          0.,FRATE,DUM)                              FRAT
      GO TO 132                                                            FRAT
  140 JPRP = 1                                                             PROPIN
        T=0.                                                               PO14
      FVAC=0.                                                              PO14
      IF(FRATE.LE.0 ) RETURN                                               FRAT
      AE = EJ*TMULT                                                        FRAT
      FVAC = FRATE *TMULT                                                  FRAT
      FRATED = FVAC                                                        FRAT
      TBURN =TIME                                                          PH1SZ
      T=FVAC                                                               PH1SZ
      IF(JPRO.EQ.1) CALL ISPIN                                             AAA
      IF(SMAX.GT.0) GO TO  145                                             FRAT
      JPRP =2                                                              FRAT
      RETURN                                                               FRAT
  145 JPRP =3                                                              FRAT
      RETURN                                                               PROPIN
```

20 OCT 72 G 01-46

```
      150 TBURN = TIME                                                  PROPIN
          MTT   =  MT                                                   PROPIN
          AE = EJ* TMULT                                                PROPIN
          IF(GMAX) 160,160,170                                          PROPIN
      160 JPRP = 2                                                      PROPIN
          GO TO 130                                                     PROPIN
      170 JPRP = 3                                                      PROPIN
          GO TO  130                                                    PROPIN
      180 CALL STPIT   (3)                                              PROPIN
          END                                                           PROPIN
```

20 OCT 72 G.01-46

SUBRØUTINE REU3

Subroutine REU3

Entry Points EQIN, SVADYS, SAVVR, SAVDYP, INTBC, BRST, BRIN, MSDISC, CORVAR.

Purpose

EQIN: Initialize state and control vectors and calculate reference point trigonometric functions.

SVADYS: Save derivatives, $\dot{y}$, at arc end points.

SAVVR: Save current state for use in cut-off refinement.

SAVDYP: Save derivatives, $\dot{y}$, at phase end point.

INTBC: Evaluate intermediate arc end constraint misses.

BRST: Save state at end of trunk to reinitialize integration for second branch.

BRIN: Evaluate first branch constraint misses and initialize state for beginning of second branch.

MSDISC: Compute mass at beginning of next arc.

CORVAR: Refine mid-integration point state estimate and store.

614

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ALPHA	α	O	Angle of attack (RAD)	/AECO3 /(	3)	BEROCO	I	ALPHA
						BLGCON	M	ALPHA
						BL2	I	ALPHA
						FNTG	O	ALPHA
						MAMECO	I	ALPHA
						MODELA	M	ALPHA
						MODELB	O	ALPHA
						REU3	O	ALPHA
						VT	I	ALPHA
CSPSR	$\cos(\psi_r)$	O	Cosine of reference azimuth	/ORBIT /(	153)	PDBC	I	CSPSR
						REU3	O	CSPSR
CSXLMR	$\cos(\rho-\rho_r)$	O	Cosine of refernece latitude	/ORBIT /(	147)	PDBC	I	CSXLMR
						REU3	O	CSXLMR
DELTAE	δ_E	O	Engine gimbal deflection angle (RAD)	/GENF /(	554)	BLGCON	M	DELTAE
						EL1	I	DELTAE
						OUT	I	DELTAE
						REU3	O	DELTAE
						VT	I	DELTAE
DT		I	Integration interval (SEC)	/GENF /(	300)	BNTG	M	DT
						FNTG	M	DT
						REU3	I	DT
						RKTA3A	I	P
						RKTB3A	I	P
						STP3	I	DT
						YREF3	O	DT
DVAR	$\dot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
GR	g_r	I	Gravitational acceleration at surface of the earth (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PAOS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOM6	I	GR
						STAU	I	GR

8 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
IPST		I	Phase counter for first nominal trajectory	/XCODES/(	167)	AST3	I	IPST
						FNTG	M	IPST
						GUI3A	I	IPST
						REU3	I	IPST
ISTART		I	Initialization and divergance flag	/XCODES/(	147)	AST3	O	ISTART
						BLGCON	O	ISTART
						BLYNE	O	ISTART
						FNTG	I	ISTART
						MODELA	O	ISTART
						PROPIN	O	ISTART
						REU3	I	ISTART
						TEST	M	ISTART
						TOPM	M	ISTART
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BST03	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
NEQF		I	Number of equations to be integrated on forward trajectory	/XCODES/(	185)	REU3	I	NEQF
						RKTA3A	I	NN
						SDINP	O	NEQF
						STAU	I	NEQF
						TOPM	O	NEQF
						TRAN3	O	NEQF
NICNB		I	Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
						ADIC3A	I	NICNB
						BNTG	I	NICNB
						REU3	I	NICNB
						SDINP	M	NICNB
						TEST	I	NICNB
						TRAN3	I	NICNB
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB
ORBI		I	Orbiter ignition arc	/SIZING/(	316)	REU3	I	ORBI
						SIZE	I	ORBI
						VEHOF	I	ORBI

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
PSIRF	ψ_r	I	Reference azimuth (DEG)	/GLOBAL/(	68)	GEINP REU3 SDINP	I I M	PSIRF PSIRF PSIRF
RDI		I	Angle to radian conversion, 01745329252	/DATA /(	3)	BLICO DER3A FNTG GUI3A MODELA MODELB PADS1 PROPB PROPIN REU3 SDINP SOMG	I I I I I I O I I I I I	RDI RDI RDI RDI RDI RDI RDI RDI RDI RDI RDI RDI
SAVBP		M	Saved state vector at branching point for initializing second brnch	/STATE3/(	689)	REU3	M	SAVBP
SNPSR	$\sin(\psi_r)$	O	Sine of reference azimuth	/ORBIT /(	152)	PDBC REU3	I O	SNPSR SNPSR
SNXLMR	$\sin(\rho-\rho_r)$	O	Sine of reference latitude	/ORBIT /(	146)	PDBC REU3	I O	SNXLMR SNXLMR
SQ		O	A synthesis data array (37,5) that contains the flyback data and some injection quantities	/SIZING/(	74)	ENVPRM FLYBKP ISPRAT PDBC PRITVA RANGE REU3 SIZE SIZEMR SIZIN STAU SUMOUT TAMPAR TAMPER THRUST TRTOSZ VEHDF WTVOL	M M I I I M O O M M I M O M M M M M	SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ SQ
SVAR	$y\|_{t=0}$	I	Array of state values at initial problem time [sd]	/GENF /(	79)	ADJUST BNTG FNTG PRMSET REU3 SDINP TEST TOPM TRTOSZ	O I I M I M I I I	SVAR SVAR SVAR SVAR SVAR SVAR SVAR SVAR SVAR
SVBV		O	Saved state vector on trial trajectory	/STATE3/(	710)	ADICB3 REU3	I O	SVBV SVBV
SVY	y_{-1}	M	State and time array at previous compute interval	/STATE3/(	236)	DTF3 REU3 YREF3	I M I	SVY SVY SVY

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
T	T	O	Thrust (LBS)	/GENF /(	411)	ACCEL	I	T
						BLGCON	M	T
						BL4	I	T
						BL6	I	T
						BL7	I	T
						BL8	I	T
						EL2	I	T
						EQUA3	O	T
						FH1	I	T
						FH2	I	T
						FH3	I	T
						FH4	I	T
						IMPUL	I	T
						OUT	I	T
						PROPB	O	T
						PROPIN	O	T
						REU3	O	T
						SDER3	I	T
TIME	t	M	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
VAR	v	M	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V
W	W	I	Weight (LBS)	/GENF /(	412)	BL5	I	W
						ENVPRM	I	W
						EQUA3	M	W
						FH3	I	W
						OUT	I	W
						PDBC	I	W
						REU3	I	W
						TRTOSZ	I	W
WDC		I	Array of drop weight per arc[sd] (LBS)	/GENF /(	89)	BNTG	I	WDC
						REU3	I	WDC
						SDINP	M	WDC

8 NOV 72 6 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
XLAMRF	ρ_r	I	Reference latitude.	(DEG)	/GLOBAL/(	4)	CRASH GEINP REU3 SDINP	M I I M	RHOO XLAMRF XLAMRF XLAMRF
YDP		O	Array of state derivatives at phase end points		/STATE3/(	327)	ADID3A REU3	I O	YDP YDP
YDS	$\dot{y}^{l}{}_{\tau-}$	O	Array of state derivatives at arc end points		/STATE3/(	507)	ADICB3 ADID3A REU3 STAU	I I O I	YDS YDS YDS YDS
YMURF	μ_r	I	Reference longitude.	(DEG)	/GLOBAL/(	5)	CRASH GEINP REU3 SDINP	M I I M	UMUO YMURF YMURF YMURF
YMXRF	ρ_r	O	Reference longitude	(RAD)	/ORBIT /(	145)	PDBC REU3	I O	YMXRF YMXRF

```
      SUBROUTINE REU3                                                      REU3
C  THIS SUBROUTINE PERFORMS VARIOUS  STATE DEPENDANT FUNCTIONS FOR ENTG  REU3
C                                                                          REU3
C         EACH  FUNCTION IS CONTAINED IN  SEPARATE  ENTRY  POINTS          COMN
      COMMON/ ORBIT/  VI,       GAMI,      PSII,      XMUI,         P,     ORBIT
     *      ECC,     AINCL,     ARGP,    ASCNOD,    SMIMAJ,    APOGEE,    ORBIT
     *  PERGEE,    ANOMLY,     CAPX,      CAPY,     ASYMP,    ENERGY,    ORBIT
     *   HMNTM,    DVIDV,     DVIDG,                                       ORBIT
     *  DVIDH ,    DVIDM,    DVIDPS,    DVIDRO,    DVIDMU,     DGIDV,    ORBIT
     *  DGIDG ,    DGIDH,    DGIDM ,    DGIDPS,    DGIDRO,     DGIDMU,   ORBIT
     *  DPIDV ,    DPIDG,    DPIDH ,    DPIDM ,    DPIDPS,     DPIDRO,   ORBIT
     *  DPIDMU,    DMIDV,    DMIDG ,    DMIDH ,    DMIDM ,     DMIDPS,   ORBIT
     *  DMIDRO,    DMIDMU,   DPDV  ,    DPDG  ,    DPDH  ,     DPDM  ,   ORBIT
     *  DPDPS ,    DPDRO ,   DPDMU ,    DECDV ,    DECDG ,     DECDH ,   ORBIT
     *  DECDM ,    DECDPS,   DECDRO,    DECDMU,    DIDV  ,     DIDG  ,   ORBIT
     *  DIDH  ,    DIDM  ,  DIDPS ,   DIDRO  ,   DIDMU  ,    DBEDV  ,    ORBIT
     *  DBEDG ,    DBEDH ,   DBEDM ,    DBEDPS,    DBEDRO,     DBEDMU,   ORBIT
     *  DNODV ,    DNODG ,   DNODH ,    DNODM ,    DNODPS,     DNODRO,   ORBIT
     *  DNODMU,    DSMDV,     DSMDG,     DSMDH,     DSMDM,    DSMDPS     ORBIT
      COMMON/ORBIT/                                                        ORBIT
     *  DSMDRO,   DSMDMU,    DAPDV,     DAPDG,     DAPDH,    DAPDM ,     ORBIT
     *  DAPDPS,   DAPDRO,    DAPDMU,    DPEDV,     DPEDG,    DPEDH ,     ORBIT
     *DPEDM     ,DPEDPS     ,DPEDRO     ,DPEDMU     ,DANDV     ,DANDG   , ORBIT
     *  DANDH ,   DANDM ,    DANDPS,    DANDRO,    DANDMU,   DCXDV ,     ORBIT
     *  DCXDG ,   DCXDH ,    DCXDM ,    DCXDPS,    DCXDRO,   DCXDMU,     ORBIT
     *  DCYDV ,   DCYDG ,    DCYDH ,    DCYDM ,    DCYDPS,   DCYDRO,     ORBIT
     *  DCYDMU,   DASDV ,    DASDG ,    DASDH ,    DASDM ,   DASDPS,     ORBIT
     *  DASDRO,   DASDMU,    DENDV ,    DENDG ,    DENDH ,   DENDM ,     ORBIT
     *  DENDPS,   DENDRO,    DENDMU,    DMODV ,    DMODG ,   DMODH ,     ORBIT
     *  DMODM ,   DMODPS,    DMODRO,    DMODMU                           ORBIT
      DIMENSION ORBPRM(18),PPO(7,18)                                       ORBIT
      EQUIVALENCE (VI,ORBPRM),(DVIDV,PPO)                                  ORBIT
      COMMON/ORBIT/ YMXRF,SNXLMR,CSXLMR,SDOWN,SCROSS,TD,TC                 ORBIT
     *,SNPSR      ,CSPSR      ,SNGI       ,CSGI       ,SPSII      ,CPSII       , ORBIT
     * STOT       ,CSI        ,SNI        ,SNGNU      ,CSAND      ,COSOMU      , ORBIT
     * SINDMU     ,THT        ,WTFUEL                                      ORBIT
      COMMON/STATE3/                                                       STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)      ,SVY(10)    , STATE3D
     *XL(9,9)    ,YDP(20,9),YOS (20,9),COSGAM     ,SINGAM     ,SAVBP(15)  , STATE3D
     *SINPSI     ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2     , STATE3D
     *SVBV   (9),OMEGA      ,OMEGAZ     ,                                  STATE3D
     *VDV        ,GDV        ,RDV        ,MDV        ,PDV        ,ODV        , STATE3D
     *UDV        ,VDG        ,GDG        ,RDG        ,PDG        ,ODG        , STATE3D
     *UDG        ,VDR        ,GDR        ,MDR        ,PDR        ,ODR        , STATE3D
     *UDR        ,VDM        ,GDM        ,MDM        ,PDM        ,VDP        , STATE3D
     *GDP        ,PDP        ,ODP        ,UDP        ,VDO        ,GDO        , STATE3D
     *PDO        ,UDO        ,HTDV       ,HTDR                               STATE3D
      REAL MDM , MDV, MDR                                                  STATE3D
      COMMON/STATE3/                                                       STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                       STATE3D
      COMMON /XCODES/                                                      XCODES
     *ITQ    (9),ICOR  (20),ITI        ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)             ,NCNST      ,NSB        ,NSAB       ,NICNB      , XCODES
     *I2OP       ,ICOP       ,IFAW       ,IFAR       ,IFB        ,IND        , XCODES
     *IOPEN      ,IPH        ,ISPH       ,ISST       ,IARC       ,ISTART     , XCODES
     *ITCT       ,ITER       ,IVAR       ,JK         ,JPS        ,JS         , XCODES
     *KOP        ,KPST       ,K          ,KST        ,NAD        ,NCASE      , XCODES
     *NCN        ,NEQB       ,NEQ        ,NOP        ,NPH        ,N          , XCODES
     *NST        ,IPST       ,IPRINT     ,ISTN       ,IPHN       ,ISTNB      , XCODES
     *IPHNB      ,IBLK1      ,IBLK2      ,ISTOP      ,ISTPP      ,L          , XCODES
     *IFOB       ,NB         ,LB         ,MB         ,NPHP       ,NPHB       , XCODES
     *NCTIN      ,NEOF ,ILAB(8),JPRP,JGII,ATT,MPIN(20),JP1,JP2,JP3           XCODES
      COMMON/DATA/                                                         DATA
     *PI         ,RAD        ,RDI        ,SC         ,UMF        ,TMPF       , DATA
     *FTNM       ,CAR        ,JOP1       ,JOP2       ,JOP3       ,JOP4         DATA
      COMMON/GENF/                                                         GENF
     *OMG(20)    ,OMGP(20,2),VARO(9)    ,TOL(9)     ,SVAR(10)   ,WOC(20)    , GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)    ,DTP        , GENF
     *DTS        ,DT         ,G          ,DFSQ       ,Q          ,QS         , GENF
     *R          ,RE         ,MACH       ,PA         ,RO         ,CS         , GENF
     *VNU        ,PAR        ,ROR        ,CSR        ,VNR        ,SUMSQ      , GENF
     *SVSQ       ,TIMEPH     ,TIMES      ,TOP        ,TOS        ,TR(9)      , GENF
     *TST(20)    ,TPH   (20),DIS(20)    ,DIP(20)    ,T          ,W          , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)   ,DIS1(20)   ,TIME       ,DMP        , GENF
     *TIMPR      ,LIFT       ,DRAG       ,TAX        ,TBURN      ,TBU(20)    , GENF
```

```
      *AE         ,FP         ,FPOLD      ,FPD        ,MACHR      ,MACHV      , GENF
      *QR         ,QV         ,FVAC       ,LIFTV      ,                         GENF
      *LIFTR      ,LIFTA      ,            DRAGV      ,DRAGR      ,DRAGA      , GENF
      *            LIFTM      ,DBR        ,DB         ,ISP        ,ISPF       , GENF
      *                        ULFT       ,ULFTV      ,ULFTR      ,ULFTA      , GENF
      *XMCG       ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM      ,CODAE      , GENF
      *CULFT      ,CT         ,CALPHA     ,CDE        ,DELTAE     ,SID        , GENF
      *COD        ,SIDAE      ,XCG        ,ZCG        ,XJ                       GENF
       COMMON / GENF  /                                                         GENF
      *XJV        ,XJR        ,GH         ,GAMMAD     ,XKG        ,XKP        , GENF
      *FRATED     ,IRATED     ,                                                 GENF
      *P1         ,P2         ,P3        ,XK1         ,XK2        ,XK3        , GENF
      *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D       ,XK3D       , GENF
      *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V       ,XK3V       , GENF
      *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P       ,XK3P       , GENF
      *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O       ,XK3O       , GENF
      *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M       ,XK3M       , GENF
      *PV         ,PG         ,PP         ,PR         ,PO         ,DPDY(3,8)    GENF
       REAL  LIFTR , LIFT , LIFTA,LIFTM ,                 MACH, MACHR,          GENF
      * ISP, ISPF, MACHV,LIFTV , IRATED                                         FRAT
       DIMENSION TPH1(10),TST1(10)                                              GENF
       EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                       GENF
       COMMON/ARCDAT/                                                           ARCDAT
      *SREF       ,EJ         ,XISP       ,TMULT      ,DTNC       ,DTPI       , ARCDAT
      *IATM       ,IMODE      ,JAER       ,JPRO       ,QMAX       ,GMAX       , ARCDAT
      *XLMAX      ,HDMAX      ,GMDOT      ,ALFMAX     ,PHMAX      ,MAEA       , ARCDAT
      *MAEB       ,MAEC       ,MAED       ,MAEE       ,MAEF       ,MAEG       , ARCDAT
      *MT         ,MISP       ,MXCG       ,MZCG       ,MWDA       ,MWDB       , ARCDAT
      *MDB        ,XCGR       ,ZCGR       ,XE         ,ZE         ,XT         , ARCDAT
      *DREF       ,MCND       ,RHOB       ,QMULT      ,REMAX                    ARCDAT
      * ,FRATE  ,  ARCD(9)                                                      RETAP
       DIMENSION  ARCDA(40)                                                     ARCDAT
       EQUIVALENCE(SREF,ARCDA)                                                  ARCDAT
       COMMON/GLOBAL/                                                           GLOBAL
      *GR         ,ER         ,OMGZ       ,XLAMRF     ,YMURF      ,LUM          GLOBAL
      *,JJOP(10) ,IFATAL      ,NARC          ,NBRAN   ,NFARC      ,ID(4)        GLOBAL
      *,KTAB(20),ITAB(20)   ,SIG    ,MAXTAB                                     GLOBAL
      *,GM        ,PSIRF,IPFLG1,       IPFLG2,IPFLG3,IPFLG4,INEQFL(20)          GLOBAL
      *,ITPSO ,KSOL      ,KGLOBL(8)                                             RETAP
       COMMON/STS/                                                              STS
      *DPAY       ,PMIN       ,WORK (20),NWDS       ,IPC     (7),NITER      ,   STS
      *MNGA(20,2),MNSP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)                  STS
       COMMON/AECO3/                                                            AECO3
      *APHO       ,APHR       ,ALPHA      ,VDA        ,GDA        ,PDA        , AECO3
      *SINA       ,COSA       ,PHIO       ,PHID       ,PHI        ,SINPHI     , AECO3
      *COSPHI     ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)   ,CDO        , AECO3
      *CDOM       ,CLO        ,FK         ,XCGM       ,ZCGM       ,CLOM       , AECO3
      *CM         ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM       ,FKM        , AECO3
      *CLAM       ,CL         ,CLA        ,CLM        ,                         AECO3
      *CD         ,CDA        ,CDM                                              AECO3
       DIMENSION  SVDQ(9)                                                       REU3
       REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                                  SIZING
       COMMON /SIZING/                                                          SIZING
C            PHASE II SIZING PARAMERERS                                         SIZING
      *TZ,       VV(3),      QP(14),    EROR,     PZ(5),      VQ,       SW(20), SIZING
      *SV(28),   SQ(37,5),  SE(11),     TLAT,     TLNG,                         SIZING
C           PHASE I SIZING PARAMERERS                                           SIZING
      *WBO,       WLOO,      DWEB,      DWEO,      TOLWT,      WPB,     TWRAT2, SIZING
      *BK1,       BK2,       BK3,       BK4,       ISIZE,      TRAFLG,  TWRATO, SIZING
      *OK1,       OK2,       OK3,       OK4,       PRFLG,      IPASS,   IPSMAX, SIZING
      *AEXIT,     TVACO,     NO,        WFO,       IDVEL,      ISPO,    ISPB,   SIZING
      *XPL,       TVACB,    NNB,        WEO,       WEB,        WO,      WLO,    SIZING
      *DVO,       DVB,       MUB,       MUO,       VSTG,       WPO              SIZING
      *,JTYP,     BECO      BSTG ,      ORBI,     ITNBW , ITNOW  ,              SIZING
      * SVOPSQ   , SVDCON  ,IHUNT     ,IOPSTG   ,ISZD(19)                       UH
C        I      INITIALIZE  STATE  AND  CONTROL VECTORS                         CONN
C               ALSO  CALCULATE   REF   LOCATION  FUNCTIONS                     CONN
       ENTRY EQIN                                                               REU3
       DO 10 I=1,NEQ                                                            REU3
    10 VAR(I)= SVAR(I+1)                                                        REU3
       DO 20 I=1,4                                                              REU3
    20 VAR(NEQ+I)=0                                                             REU3
       YMXRF=YMURF*RDI                                                          REU3
       XLAXRF = XLAMRF *RDI                                                     REU3
       SNXLMR= SIN(XLAXRF)                                                      REU3
```

```
      CSXLMR= COS(XLAXRF)                                               REU3
      SNPSR=SIN(PSIRF*RDI)                                              REU3
      CSPSR=COS(PSIRF*RDI)                                              REU3
      ALPHA=0.                                                          P014
      DELTAE=0.                                                         P014
      T=1.E6                                                            P014
      RETURN                                                            REU3
C                                                                       COMN
C        II   SAVE  DERIVATIVES  AT  ARC  END   POINT                   COMN
      ENTRY SVADYS                                                      REU3
      DO 30  I=1,NEQ                                                    REU3
   30 YDS(IARC,I) = DVAR(I)                                             REU3
      RETURN                                                            REU3
C                                                                       COMN
C        III   SAVE  STATE  FOR  CUT-OFF  REFINEMENT  (SEE  YREF)       COMN
      ENTRY  SAVVR                                                      REU3
      DO  40  I=1,NEQ                                                   REU3
   40 SVY(I+1)=VAR(I)                                                   REU3
      SVY(1)=TIME                                                       REU3
      RETURN                                                            REU3
C                                                                       COMN
C        IV   SAVE  DERIVATIVES  AT  PHASE  END  POINT                  COMN
      ENTRY  SAVDYP                                                     REU3
      DO 50 I=1,NEQ                                                     REU3
   50 YDP(IPST,I)= DVAR(I)                                              REU3
      RETURN                                                            REU3
C                                                                       COMN
C        V    EVALUATE  INTERMEDIATE  CONSTRAINT  MISS                  COMN
C             ACCOUNTING  FOR  WHETHER  ON  TRIAL  OR  STARTING  TRAJ.  COMN
      ENTRY INTBC                                                       REU3
      IF(ISTART GT.1) GO TO 60                                          REU3
      CALL CONIN(1,NICNB)                                               REU3
      RETURN                                                            REU3
   60 CALL CON(1,NICNB)                                                 REU3
      RETURN                                                            REU3
C                                                                       COMN
C        VI     SAVE   STATE  AT  END  OF  TRUNK  TO  REINITIALIZE      COMN
C               INTEGRATION  FOR  SECOND  BRANCH                        COMN
      ENTRY BRST                                                        REU3
      NPHB=IPST                                                         REU3
      DO 70  I=1,NEQF                                                   REU3
   70 SAVBP(I+1)= VAR(I)                                                REU3
      SAVBP(1) = TIME                                                   REU3
      RETURN                                                            REU3
C                                                                       COMN
C        VI-B   EVALUATE  BRANCH 1   CONSTRAINT  MISSES  AND  INITIALIZE COMN
C               STATE   TO  START  BRANCH  2  INTEGRATION               COMN
      ENTRY  BRIN                                                       REU3
      NPHP=IPST+1                                                       REU3
      IF( ISTART GT.1) GO TO 80                                         REU3
      CALL CONIN(1,NICNB)                                               REU3'
      GO TO 90                                                          REU3
   80 CONTINUE                                                          REU3
      CALL CON(1,NICNB)                                                 REU3
   90 DO  110  I=1,NEQF                                                 REU3
  100 SVBV(I)= VAR(I)                                                   REU3
  110 VAR(I)= SAVBP(I+1)                                                REU3
      TIME= SAVBP(1)                                                    REU3
      IF(WDC(IARC-1).NE.0.) RETURN                                      DS
      VAR(4) = WDC(NSB)/GR                                              REU3
      RETURN                                                            REU3
C                                                                       COMN
C        VII      COMPUTE  MASS  AT  START  OF NEXT  ARC                COMN
      ENTRY MSDISC                                                      REU3
C            WEIGHT  DROP LOGIC                                         PH1SZ
C         VII-A    IS  WT  DROP  COMPUTATION  FLAGGED  (PHASE II SIZING) COMN
C          GO TO  VII-E   ELSE  VII-B                                   COMN
      IF(WDC(IARC-1).EQ.-99999999.) GO  TO  135                         PH1SZ
C          VII-B   IS  NEXT ARC  WT. JUST  INITIALIZED, GO TO  VII-C    COMN
C           ELSE  VII-D                                                 COMN
      IF(WDC(IARC-1))120,130,130                                        REU3
C            VII-C    INITIALIZE  MASS FOR NEXT ARC                     COMN
  120 VAR(4) =-WDC(IARC-1)/GR                                           REU3
```

627

```
      I = ORBI                                                   OS
      IF( I EQ IARC) SQ(3,5) = VAR(4)*GR                         OS
      RETURN                                                     REU3
  130 CONTINUE                                                   REU3
C         VII-D    DROP  WT    AND  CALC. MASS                   COMN
      VAR(4) =VAR(4) -WDC(IARC-1)/GR                             REU3
      I = ORBI                                                   OS
      IF( I EQ.IARC) SQ(3,5) = VAR(4)*GR                         OS
      RETURN                                                     REU3
C         VII-E    COMPUTE  PROPPELLANT  WT.  DEPENDANT  DROP  WT.  COMN
C                  AND  CALC  MASS                               COMN
  135 WP1 =  SVAR(5)*GR - W                                      PH1SZ
      CALL  WTORP(WP1,WD,WDP,1)                                  PH1SZ
      VAR(4)= VAR(4) - WD/GR                                     PH1SZ
      SQ(3,5)= VAR(4)*GR                                         OS
      RETURN                                                     PH1SZ
C                                                                COMN
C        VIII    REFINE  MID-INTEGRATION STATE ESTIMATE  AND  STORE  COMN
      ENTRY CORVAR                                               REU3
      DO 140 I=1,NEQ                                             REU3
  140 SVDQ(I)=DVAR(I)                                            REU3
      CALL DER                                                   REU3
      DO 150 I=1,NEQ                                             REU3
  150 VAR(I)=SVY(I+1)+(SVDQ(I)+DVAR(I))*DT*.25                   REU3
      CALL ASTO                                                  REU3
      RETURN                                                     REU3
      END                                                        REU3
```

20 OCT 72 6 01-46

623

SUBRØUTINE
RKTA3A

Subroutine RKTA3A

Entry Point RKUTTA

Purpose

Subroutine RKTA3A is the utility RUNGE-KUTTA integration routine used for integrating the foward trajectory.

Description

This Runge-kutta package employs a four-cycle method with two derivatives at the mid-integral and two at the end interval. This routine is called from FNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DY	$\dot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
F		M	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A	I	VARL
						ADICB3	O	VARL
						ADIC3A	O	VARL
						PROPIN	I	ZZ
						RKTA3A	M	F
						RKTB3A	M	Y
						STVRL3	O	VARL
						TRAN3	M	VARL
J		M	Integration routine flag tells which derivative evaluation in Runge-Kutta cycle	/XCODES/(	151)	ADIC3A	M	JK
						BNTG	I	JK
						MODELA	I	JK
						PAY02	M	JK
						RKTA3A	M	J
						RKTB3A	M	J
L		M	Integration traffic control flag L = 1 means evaluate derivatives = 2 check cut-off = 3 print or cut-off detected	/XCODES/(	177)	BNTG	M	L
						FNTG	M	L
						OUT	I	L
						RKTA3A	M	L
						RKTB3A	M	L
						SDINP	M	L
NN		I	Number of equations to be integrated on forward trajectory	/XCODES/(	185)	REU3	I	NEQF
						RKTA3A	I	NN
						SDINP	O	NEQF
						STAU	I	NEQF
						TOPM	O	NEQF
						TRAN3	O	NEQF
P		I	Integration interval (SEC)	/GENF /(	300)	BNTG	M	DT
						FNTG	M	DT
						REU3	I	DT
						RKTA3A	I	P
						RKTB3A	I	P
						STP3	I	DT
						YREF3	O	DT
TP		I	Trajectory print time	/GENF /(	495)	BNTG	O	TIMPR
						FNTG	M	TIMPR
						RKTA3A	I	TP
						RKTB3A	I	TP
TT	t	M	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME

629

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
V	v	M	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	Y
							STP3	I	VAR
							TOPM	O	KWOW
							VREF3	M	V

```
      SUBROUTINE RKTA3A                                         RKTA3A
      DIMENSION PR(3),TH(3)                                     RKTA3A
      COMMON/STATE3/ Y(14),DY(14), F(42),DELY(28)               RKTA3A
      COMMON/GENF/ GOO (307)                                    RKTA3A
      EQUIVALENCE (GOO(300),P),(GOO(493),TT),(GOO(495),TP)      IO
      COMMON/XCODES/ IIJ(43)                                    RKTA3A
      DIMENSION III(1)                                          RKTA3A
      EQUIVALENCE(IIJ(136),III)                                 RKTA3A
      EQUIVALENCE (III(50),NN),(III(42),L),(III(16),J)          RKTA3A
      ENTRY RKUTTA                                              RKTA3A
      T = TT                                                    RKTA3A
      GO TO (10,40,140,250),L                                   RKTA3A
   10 IG = IG                                                   RKTA3A
      IF(IG.EQ 2) 150,20                                        RKTA3A
   20 DO 30 I = 1,N                                             RKTA3A
      K = 2*N + I                                               RKTA3A
   30 F(K) = Y(I)                                               RKTA3A
      L = 2                                                     RKTA3A
      GO TO 270                                                 RKTA3A
   40 T2 = TP - T                                               RKTA3A
      IF(TP.NE 0.) 50,60                                        RKTA3A
   50 IF(ABS(T2/TP)-1 E-7) 130,60,60                            RKTA3A
   60 H1 = ABS(T2)                                              RKTA3A
      IF(TP.EQ.0.) 70,80                                        RKTA3A
   70 IF(H1-1 E-7) 130,80,80                                    RKTA3A
   80 IF(H1 - ABS(1.01*P ))90,90,120                            RKTA3A
   90 IF(T2)100,110,110                                         RKTA3A
  100 DT = -H1                                                  RKTA3A
      GO TO 120                                                 RKTA3A
  110 DT = H1                                                   RKTA3A
  120 IG = 2                                                    RKTA3A
      J = 1                                                     RKTA3A
      GO TO 150                                                 RKTA3A
  130 T = TP                                                    RKTA3A
      L = 3                                                     RKTA3A
      J = I                                                     RKTA3A
      GO TO 270                                                 RKTA3A
  140 IG = 2                                                    RKTA3A
      DT = P                                                    RKTA3A
  150 DO 200 I = 1,N                                            RKTA3A
      YK = DY(I)*DT                                             RKTA3A
      GO TO(160,170,170,180),J                                  RKTA3A
  160 F(I) = YK                                                 RKTA3A
      GO TO 190                                                 RKTA3A
  170 F(I) = F(I) + 2.*YK                                       RKTA3A
      GO TO 190                                                 RKTA3A
  180 F(I) = F(I) + YK                                          RKTA3A
      GO TO 200                                                 RKTA3A
  190 K = 2*N + I                                               RKTA3A
      Y(I) = F(K) + PR(J)*YK                                    RKTA3A
  200 CONTINUE                                                  RKTA3A
      IF(J-4)210,230,230                                        RKTA3A
  210 T = T+TH(J)*DT                                            RKTA3A
      J = J + 1                                                 RKTA3A
  220 L = 1                                                     RKTA3A
      GO TO 270                                                 RKTA3A
  230 DO 240 I = 1,N                                            RKTA3A
      M =    N + I                                              RKTA3A
      DELY(I) = F(I)/6.    + DELY(I)                            RKTA3A
  240 Y(I) = F(M) + DELY(I)                                     RKTA3A
      IG = 1                                                    RKTA3A
      GO TO 220                                                 RKTA3A
  250 N = NN                                                    RKTA3A
      PR(1)=PR(2)=TH(1)=TH(3)=.5                                RKTA3A
```

20 OCT 72 G.01-46

```
      PR(3)=1.                                                          RKTA3A
      TH(2)=0.                                                          RKTA3A
      DT = P                                                            RKTA3A
      IG = 1                                                            RKTA3A
      J = 1                                                             RKTA3A
      L = I                                                             RKTA3A
      DO 260  I = 1,N                                                   RKTA3A
      M =    N + I                                                      RKTA3A
      DELY(I) = 0.                                                      RKTA3A
  260 F(M) = Y(I)                                                       RKTA3A
  270 TT = T                                                            RKTA3A
      RETURN                                                            RKTA3A
      END                                                               RKTA3A
```

20 OCT 72 6.01-46

629

SUBRØUTINE RKTB3A

Subroutine RKTB3A

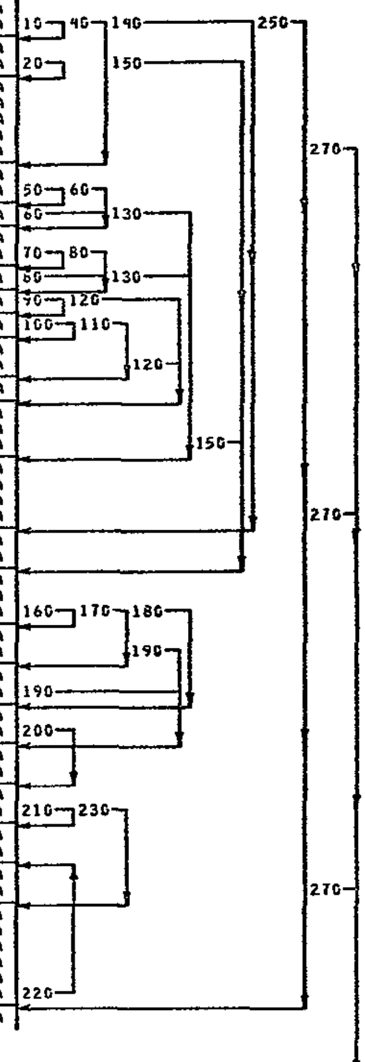
Entry Point RKUTTB

Purpose

Subroutine RKTB3A is the utility RUNGE-KUTTA integration routine used for integrating the adjoint solution.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DY		I	Array of derivatives for adjoint integration	/STATE3/(	128)	ADEQ3A	O	DVARL
						ADICB3	O	DVARL
						ADIC3A	O	DVARL
						RKTB3A	I	DY
J		M	Integration routine flag tells which derivative evaluation in Runge-Kutta cycle	/XCODES/(	151)	ADIC3A	M	JK
						BNTG	I	JK
						MODELA	I	JK
						PAYO2	M	JK
						RKTA3A	M	J
						RKTB3A	M	J
L		M	Integration traffic control flag L = 1 means evaluate derivatives = 2 check cut-off = 3 print or cut-off detected	/XCODES/(	177)	BNTG	M	L
						FNTG	M	L
						OUT	I	L
						RKTA3A	M	L
						RKTB3A	M	L
						SDINP	M	L
NB		I	Extent of integration set during adjoints on branch problem	/XCODES/(	179)	ADEQ3A	I	NB
						ADICB3	M	NB
						ADIC3A	M	NB
						BNTG	O	NB
						RKTB3A	I	NB
						STVRL3	I	NB
NN		I	Number of integrated quantities during adjoint solution	/XCODES/(	161)	ADICB3	I	NEQB
						ADIC3A	M	NEQB
						RKTB3A	I	NN
P		I	Integration interval (SEC)	/GENF /(	300)	BNTG	M	DT
						FNTG	M	DT
						REU3	I	DT
						RKTA3A	I	P
						RKTB3A	I	P
						STP3	I	DT
						YREF3	O	DT
TP		I	Trajectory print time	/GENF /(	495)	BNTG	O	TIMPR
						FNTG	M	TIMPR
						RKTA3A	I	TP
						RKTB3A	I	TP
TT	t	M	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME
Y		M	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A	I	VARL
						ADICB3	O	VARL
						ADIC3A	O	VARL
						PROPIN	I	ZZ
						RKTA3A	M	F
						RKTB3A	M	Y
						STVRL3	O	VARL
						TRAN3	M	VARL

```
      SUBROUTINE RKTB3A                                               RKTB3A
      DIMENSION PR(3),TH(3)                                           RKTB3A
      COMMON/ STATE3/ GOO(28), Y(99),DY(99)                           RKTB3A
      DIMENSION  F(297), DELY(198)                                    RKTB3A
      COMMON/GENF/ZOO(307)                                            RKTB3A
      EQUIVALENCE (ZOO(300),P),(ZOO(493),TT),(ZOO(495),TP)            IO
      COMMON/XCODES/ IIJ(44)                                          RKTB3A
      DIMENSION III(1)                                                RKTB3A
      EQUIVALENCE(IIJ(136),III)                                       RKTB3A
      EQUIVALENCE(III(26),NN),(III(44),NB),(III(42),L),(III(16),J)    RKTB3A
      ENTRY  RKUTTB                                                   RKTB3A
      T = TT                                                          RKTB3A
      GO TO (10,40,140,250),L                                         RKTB3A
   10 IG = IG                                                         RKTB3A
      IF(IG EQ.2) 150,20                                              RKTB3A
   20 DO 30 I =NB,N                                                   RKTB3A
      K = 2*N + I                                                     RKTB3A
   30 F(K) = Y(I)                                                     RKTB3A
      L = 2                                                           RKTB3A
      GO TO 270                                                       RKTB3A
   40 T2 = TP - T                                                     RKTB3A
      IF(TP.NE 0.) 50,60                                              RKTB3A
   50 IF(ABS(T2/TP)-1 E-7) 130,60,60                                  RKTB3A
   60 H1 = ABS(T2)                                                    RKTB3A
      IF(TP.EQ.0.) 70,80                                              RKTB3A
   70 IF(H1-1.E-7) 130,80,80                                          RKTB3A
   80 IF(H1 - ABS(1.01*P ))90,90,120                                  RKTB3A
   90 IF(T2)100,110,110                                               RKTB3A
  100 DT = -H1                                                        RKTB3A
      GO TO 120                                                       RKTB3A
  110 DT = H1                                                         RKTB3A
  120 IG = 2                                                          RKTB3A
      J = 1                                                           RKTB3A
      GO TO 150                                                       RKTB3A
  130 T = TP                                                          RKTB3A
      L = 3                                                           RKTB3A
      J = 1                                                           RKTB3A
      GO TO 270                                                       RKTB3A
  140 IG = 2                                                          RKTB3A
      DT = P                                                          RKTB3A
  150 DO 200 I =NB,N                                                  RKTB3A
      YK = DY(I)*DT                                                   RKTB3A
      GO TO(160,170,170,180),J                                        RKTB3A
  160 F(I) = YK                                                       RKTB3A
      GO TO 190                                                       RKTB3A
  170 F(I) = F(I) + 2.*YK                                             RKTB3A
      GO TO 190                                                       RKTB3A
  180 F(I) = F(I) + YK                                                RKTB3A
      GO TO 200                                                       RKTB3A
  190 K = 2*N + I                                                     RKTB3A
      Y(I) = F(K) + PR(J)*YK                                          RKTB3A
  200 CONTINUE                                                        RKTB3A
      IF(J-4)210,230,230                                              RKTB3A
  210 T = T+TH(J)*DT                                                  RKTB3A
      J = J + 1                                                       RKTB3A
  220 L = 1                                                           RKTB3A
      GO TO 270                                                       RKTB3A
  230 DO 240 I =NB,N                                                  RKTB3A
      M =    N + I                                                    RKTB3A
      DELY(I) = F(I)/6.    + DELY(I)                                  RKTB3A
  240 Y(I) = F(M) + DELY(I)                                           RKTB3A
      IG = 1                                                          RKTB3A
      GO TO 220                                                       RKTB3A
  250 N = NN                                                          RKTB3A
```



20 OCT 72 6.01-46

623

```
      PR(1)=PR(2)=TH(1)=TH(3)=.5                                        RKTB3A
      PR(3)=1.                                                          RKTB3A
      TH(2)=0.                                                          RKTB3A
      DT = P                                                            RKTB3A
      IG = 1                                                            RKTB3A
      J = 1                                                             RKTB3A
      L = 1                                                             RKTB3A
      DO 260  I =NB,N                                                   RKTB3A
      M =    N + I                                                      RKTB3A
      DELY(I) = 0.                                                      RKTB3A
  260 F(M) = Y(I)                                                       RKTB3A
  270 TT = T                                                            RKTB3A
      RETURN                                                            RKTB3A
      END                                                               RKTB3A
```

20 OCT 72 6.01-46

634

SUBRØUTINE SDER3

SDER3

Entry Point SDER

Purpose

Subroutine SDER3 computes the integrands of ideal velocity and velocity losses during the solution trajectory.

636

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AE	A_{exit}	I	Total nozzle exit area		/GENF	/(520)	ACCEL	I	AE
							FH2	I	AE
							IMPUL	I	AE
							PROPB	O	AE
							PROPIN	O	AE
							SDER3	I	AE
CODAE	$\cos(\alpha-\delta_E)$	I	See symbol		/GENF	/(549)	ACCEL	I	CODAE
							BL4	I	CODAE
							BL6	I	CODAE
							BL7	I	CODAE
							BL8	I	CODAE
							FH3	I	CODAE
							SDER3	I	CODAE
							VT	O	CODAE
DB	D_b	I	Base drag	(LBS)	/GENF	/(537)	ACCEL	I	DB
							BL4	I	DB
							BL6	I	DB
							BL7	I	DB
							BL8	I	DB
							EQUA3	I	DB
							FH3	I	DB
							OUT	I	DB
							SDER3	I	DB
							VT	I	DB
DRAG	D	I	Aerodynamic drag	(LBS)	/GENF	/(497)	ACCEL	I	DRAG
							BL5	I	DRAG
							BL7	I	DRAG
							BL8	I	DRAG
							ENVPRM	I	DRAG
							FH3	I	DRAG
							OUT	I	DRAG
							PROPB	O	DRAG
							PROPIN	O	DRAG
							SDER3	I	DRAG
							VT	M	DRAG
DVAR	$\dot{y}$	O	State vector derivatives in steepest descent module		/STATE3/	(15)	ADICB3	M	DVAR
							ADIC3A	I	DVAR
							ADID3A	M	DVAR
							DER3A	O	VD
							DTF3	I	VT
							ENVPRM	I	DVAR
							PDBC	I	VD
							PROPIN	O	DVAR
							REU3	I	DVAR
							RKTA3A	I	DY
							SDER3	O	DVAR
							STP3	I	DVAR
							YREF3	I	DVAR
							YREF3	I	VT
FVAC		I	Total vacuum thrust [rocket]	(LBS)	/GENF	/(528)	ACCEL	I	FVAC
							EQUA3	M	FVAC
							FH2	I	FVAC
							IMPUL	M	FVAC
							PROPB	M	FVAC
							PROPIN	M	FVAC
							SDER3	I	FVAC
G	g	I	Gravitational attraction	(FT/SEC**2)	/GENF	/(301)	BL4	I	G
							BL7	I	G
							BL8	I	G
							DER3A	I	G
							EQUA3	M	G
							MODELA	I	G
							MODELB	I	G
							PDY3A	I	G
							SDER3	I	G
							SDINP	M	G

637

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
M	m	I	Mass		/STATE3/(	4)	ACCEL	I	M
							BL4	I	M
							BL8	I	M
							EQUA3	I	M
							OUT	I	M
							SDER3	I	M
NEQ		I	Number of integrated states		/XCODES/(	162)	ADICB3	I	NEQ
							ADIC3A	I	NEQ
							ADID3A	I	NEQ
							AGETB3	I	NEQ
							AST3-	I	NEQ
							BGET3	I	NEQ
							BSTO3	I	NEQ
							MTX3A	I	NEQ
							OUT	I	NEQ
							REU3	I	NEQ
							SDER3	I	NEQ
							SDINP	M	NEQ
							TOPM	I	NEQ
							TRAN3	I	NEQ
							YREF3	I	NEQ
PA	p_a	I	Atmospheric pressure	(PSF)	/GENF /(	308)	EQUA3	M	DZM
							FH2	I	PA
							IMPUL	I	PA
							OUT	I	PA
							PDBC	I	PA
							SDER3	I	PA
SINGAM	$\sin(\gamma)$	I	See symbol		/STATE3/(	688)	BL4	I	SINGAM
							BL7	I	SINGAM
							BL8	I	SINGAM
							DER3A	I	SINGAM
							EQUA3	O	SINGAM
							MODELA	I	SINGAM
							MODELB	I	SINGAM
							PDBC	I	SINGAM
							PDY3A	I	SINGAM
							SDER3	I	SINGAM
T	T	I	Thrust	(LBS)	/GENF /(	411)	ACCEL	I	T
							BLGCON	M	T
							BL4	I	T
							BL6	I	T
							BL7	I	T
							BL8	I	T
							EL2	I	T
							EQUA3	O	T
							FH1	I	T
							FH2	I	T
							FH3	I	T
							FH4	I	T
							IMPUL	I	T
							OUT	I	T
							PROPB	O	T
							PROPIN	O	T
							REU3	O	T
							SDER3	I	T

```
      SUBROUTINE  SDER3                                                 SDER3
C                                                                       COMN
C        IDEAL  VELOCITY  AND  VELOCITY  LOSS   INTEGRANDS              COMN
C                                                                       COMN
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10) , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                 STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                            STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM  . ,                                STATE3D
      COMMON/STS/                                                       STS
     *DPAY       ,PMIN      ,WORK (20),NWDS      ,IPC    (7),NITER     , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)           STS
      COMMON/ARCDAT/                                                    ARCDAT
     *SREF       ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI     , ARCDAT
     *IATM       ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX     , ARCDAT
     *XLMAX      ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA     , ARCDAT
     *MAEB       ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG     , ARCDAT
     *MT         ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB     , ARCDAT
     *MDB        ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT       , ARCDAT
     *DREF       ,MCND      ,RHOB      ,QMULT     ,REMAX                ARCDAT
     * ,FRATE    , ARCD(9)                                              RETAP
      DIMENSION  ARCDA(40)                                              ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                           ARCDAT
      COMMON/AECO3/                                                     AECO3
     *APHO       ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      , AECO3
     *SINA       ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI   , AECO3
     *COSPHI     ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO      , AECO3
     *CDOM       ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM     , AECO3
     *CM         ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM      , AECO3
     *CLAM       ,CL        ,CLA       ,CLM       ,                    AECO3
     *CD         ,CDA       ,CDM                                        AECO3
      COMMON/GENF/                                                      GENF
     *DMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COT1(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                     GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGA     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                   GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED     ,                                           GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XV3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8) GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPHI(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPHI),(TLS1,TST1)                                GENF
      EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,  EQUV3
     *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2), EQUV3
     *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD), EQUV3
```

639

```
      *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)            EQUV3
       REAL M,MU,MD                                                      EQUV3
       COMMON/GLOBAL/                                                    GLOBAL
      *GR         ,ER         ,OMGZ      ,XLAMRF     ,YMURF      ,LUM    GLOBAL
      *,JJOP(10)  ,IFATAL     ,NARC        ,NBRAN     ,NFARC     ,ID(4)  GLOBAL
      *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                               GLOBAL
      *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)    GLOBAL
      *,ITPSO ,KSOL     ,KGLOBL(8)                                       RETAP
       COMMON /XCODES/                                                   XCODES
      *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
      *JST   (20)          ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
      *I20P       ,ICOP       ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
      *IOPEN      ,IPH        ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
      *ITCT       ,ITER       ,IVAR      ,JK        ,JPS       ,JS       , XCODES
      *KOP        ,KPST       ,K         ,KST       ,NAD       ,NCASE    , XCODES
      *NCN        ,NEQB       ,NEQ       ,NOP       ,NPH       ,N        , XCODES
      *NST        ,IPST       ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
      *IPHNB      ,IBLK1      ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
      *IFOB       ,NB         ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
      *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3       XCODES
       ENTRY SDER                                                        SDER3
       IF(T EQ.0) GO  TO 10                                              AAA      10
C                DRAG   LOSS                                             COMN
       DVAR(NEQ+2) = (DRAG+OB*CODAE)/M                                   PH1SZ
C               IDEAL  VELOCITY                                          COMN
       DVAR(NEQ+1) =(T +AE+PA)/M                                         SDER3
C                BACK-PRESSURE  LOSS                                     COMN
       DVAR(NEQ+3)= AE*PA/M*CODAE                                        PH1SZ
C                GRAVITY  LOSS                                           COMN
       DVAR(NEQ+4)=G*SINGAM                                              PH1SZ
C                MISALIGNMENT  LOSS                                      COMN
       DVAR(NEQ+5) = FVAC*(1.-CODAE)/M                                   PH1SZ
       RETURN                                                            SDER3
   10 DO 20 I=1,5                                                        AAA
   20 DVAR(NEQ+I) = 0.                                                   AAA
       RETURN                                                            AAA
       END                                                               SDER3
```

646

SUBRØUTINE SDINP

SDINP

Purpose

SDINP scans and interprets input data.

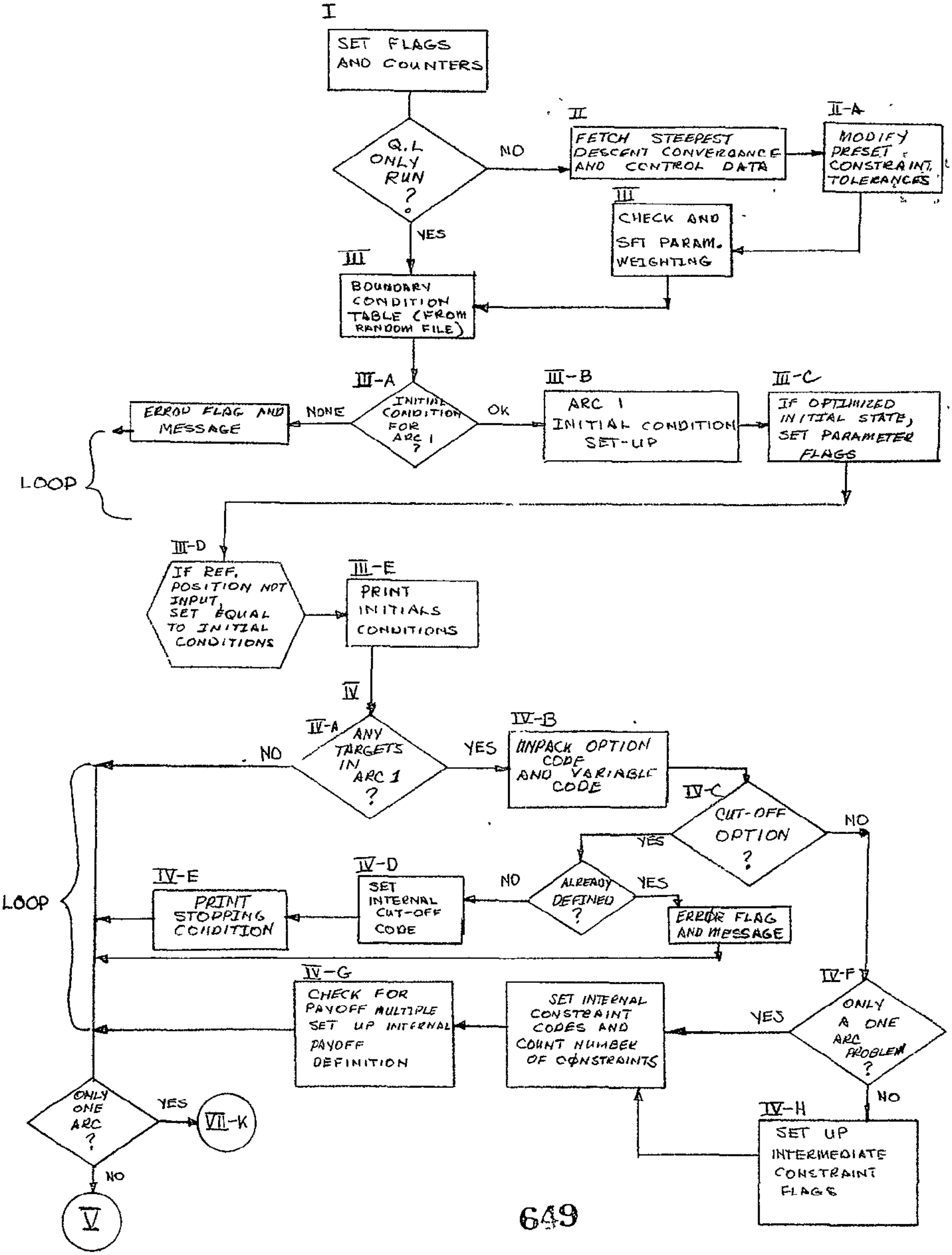
Description

SDINP is an overlay called by the Steepest Descent executive routine, TOPM. The bulk of its coding is related to converting input data into internally used data and flags for the steepest descent module; however, it serves also to check boundary condition data for the quasi-linearization module and print out this data.

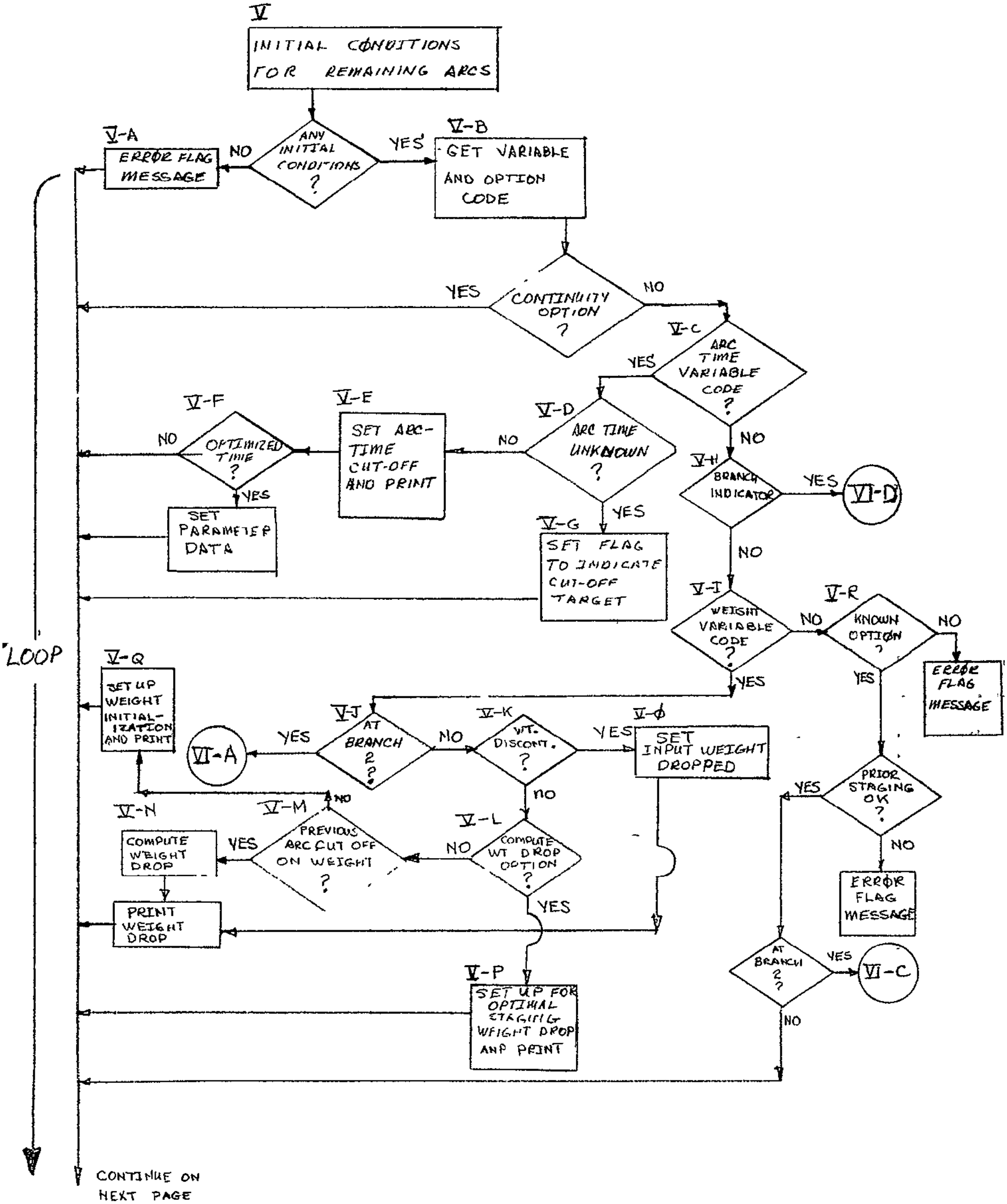
Aside from boundary condition data, SDINP processes the following type of data for the Steepest Descent program.

1. Updates of present constraint tolerances and parameter weighting factors
2. Tape starting solution initialization.
3. Solution control modes
4. Starting nominal control modes and tables (no tape solution given)
5. Steepest descent convergance data
6. Steepest descent storage flags and counters.

SDINP

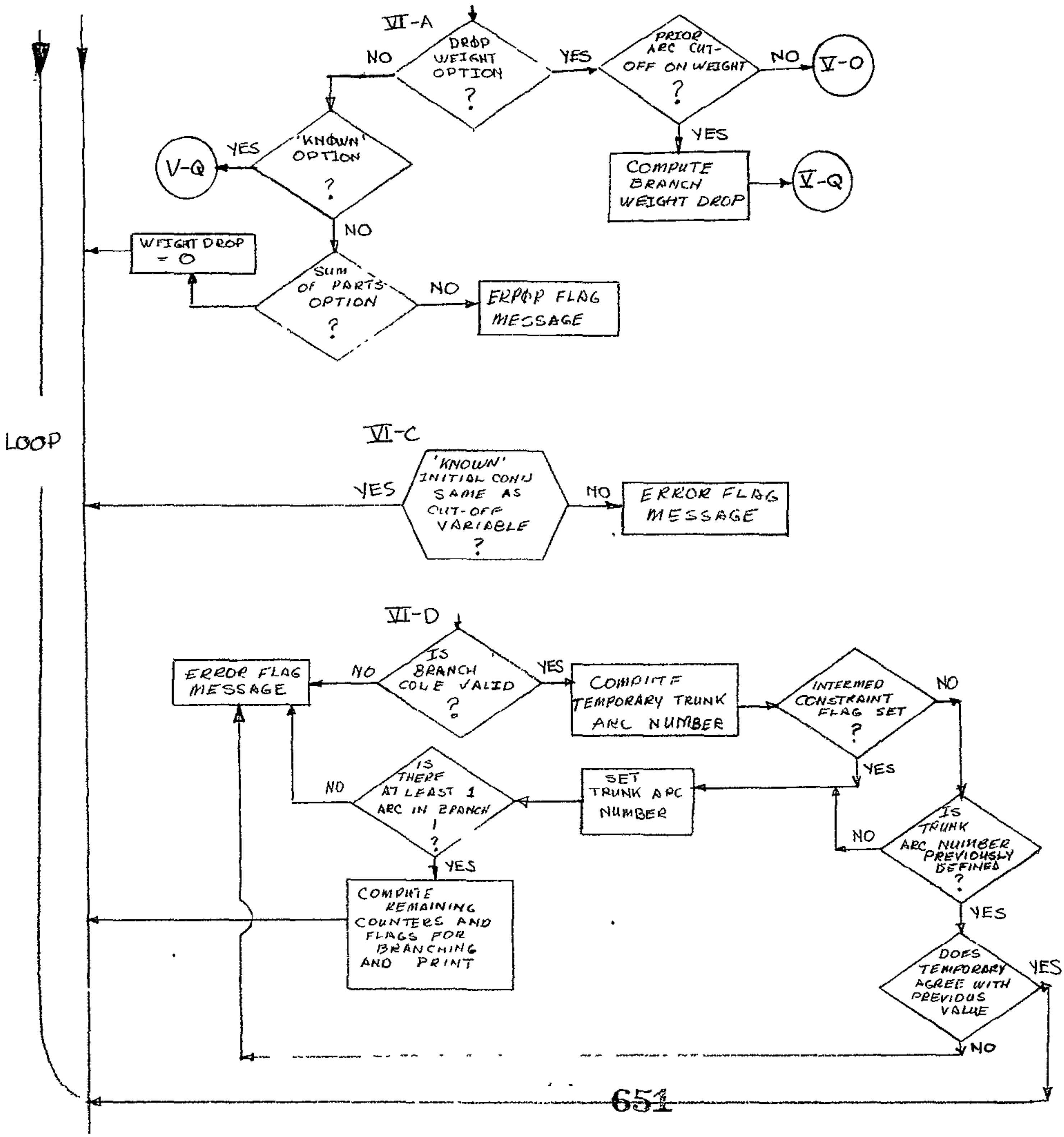

I
SET FLAGS AND COUNTERS
Q.L ONLY RUN ?
NO
YES
II
FETCH STEEPEST DESCENT CONVERGANCE AND CONTROL DATA
II-A
MODIFY PRESET CONSTRAINT TOLERANCES
III
CHECK AND SET PARAM. WEIGHTING
III
BOUNDARY CONDITION TABLE (FROM RANDOM FILE)
III-A
INITIAL CONDITION FOR ARC 1 ?
NONE
OK
ERROR FLAG AND MESSAGE
LOOP
III-B
ARC 1 INITIAL CONDITION SET-UP
III-C
IF OPTIMIZED INITIAL STATE, SET PARAMETER FLAGS
III-D
IF REF. POSITION NOT INPUT, SET EQUAL TO INITIAL CONDITIONS
III-E
PRINT INITIALS CONDITIONS
IV
IV-A
ANY TARGETS IN ARC 1 ?
NO
YES
IV-B
UNPACK OPTION CODE AND VARIABLE CODE
IV-C
CUT-OFF OPTION ?
NO
YES
ALREADY DEFINED ?
NO
YES
ERROR FLAG AND MESSAGE
IV-D
SET INTERNAL CUT-OFF CODE
IV-E
PRINT STOPPING CONDITION
LOOP
IV-F
ONLY A ONE ARC PROBLEM ?
YES
NO
SET INTERNAL CONSTRAINT CODES AND COUNT NUMBER OF CONSTRAINTS
IV-G
CHECK FOR PAYOFF MULTIPLE SET UP INTERNAL PAYOFF DEFINITION
IV-H
SET UP INTERMEDIATE CONSTRAINT FLAGS
ONLY ONE ARC ?
YES
NO
VII-K
V

SDINP (CONTINUED)



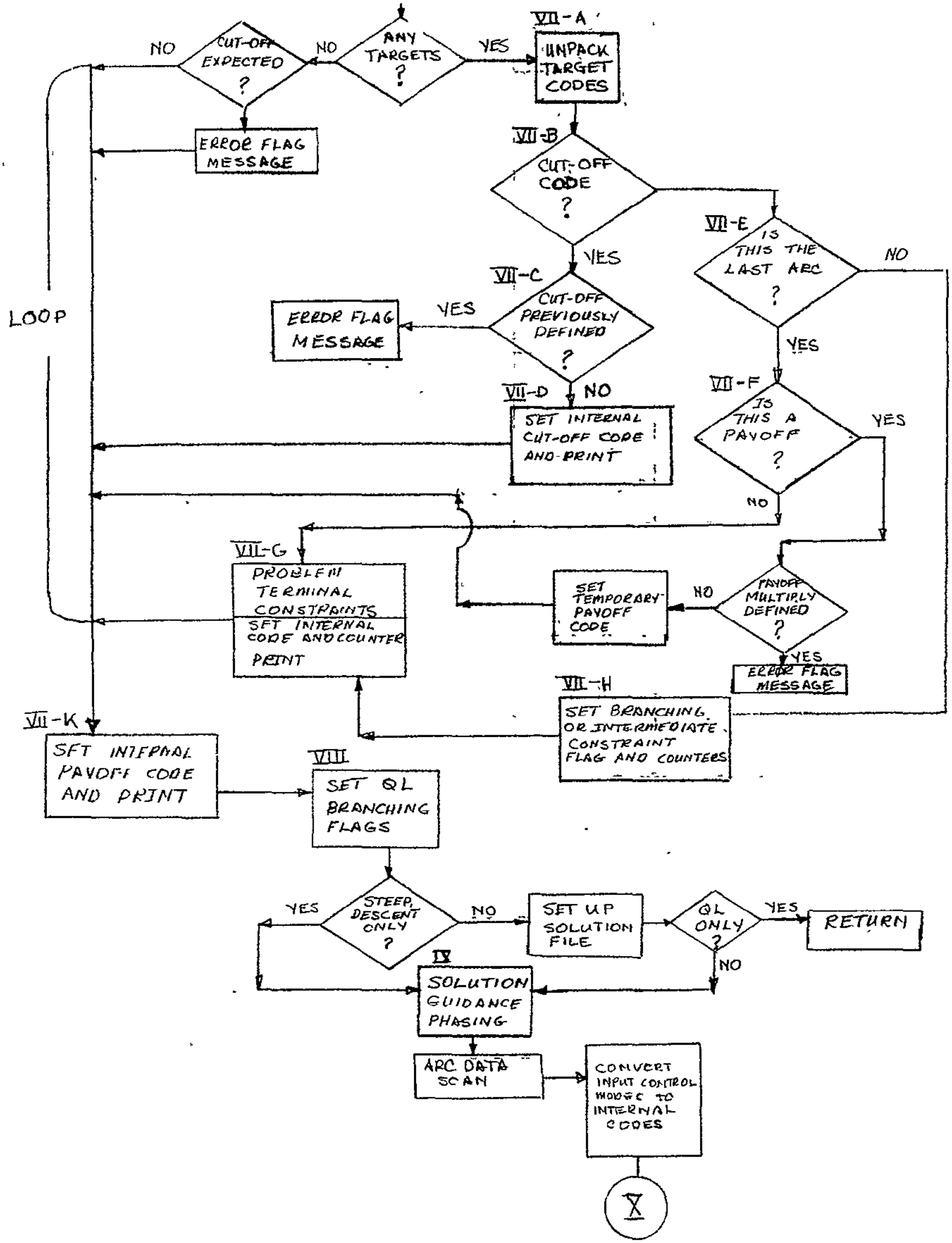
SDINP (CONTINUED)

VI BRANCH LOGIC



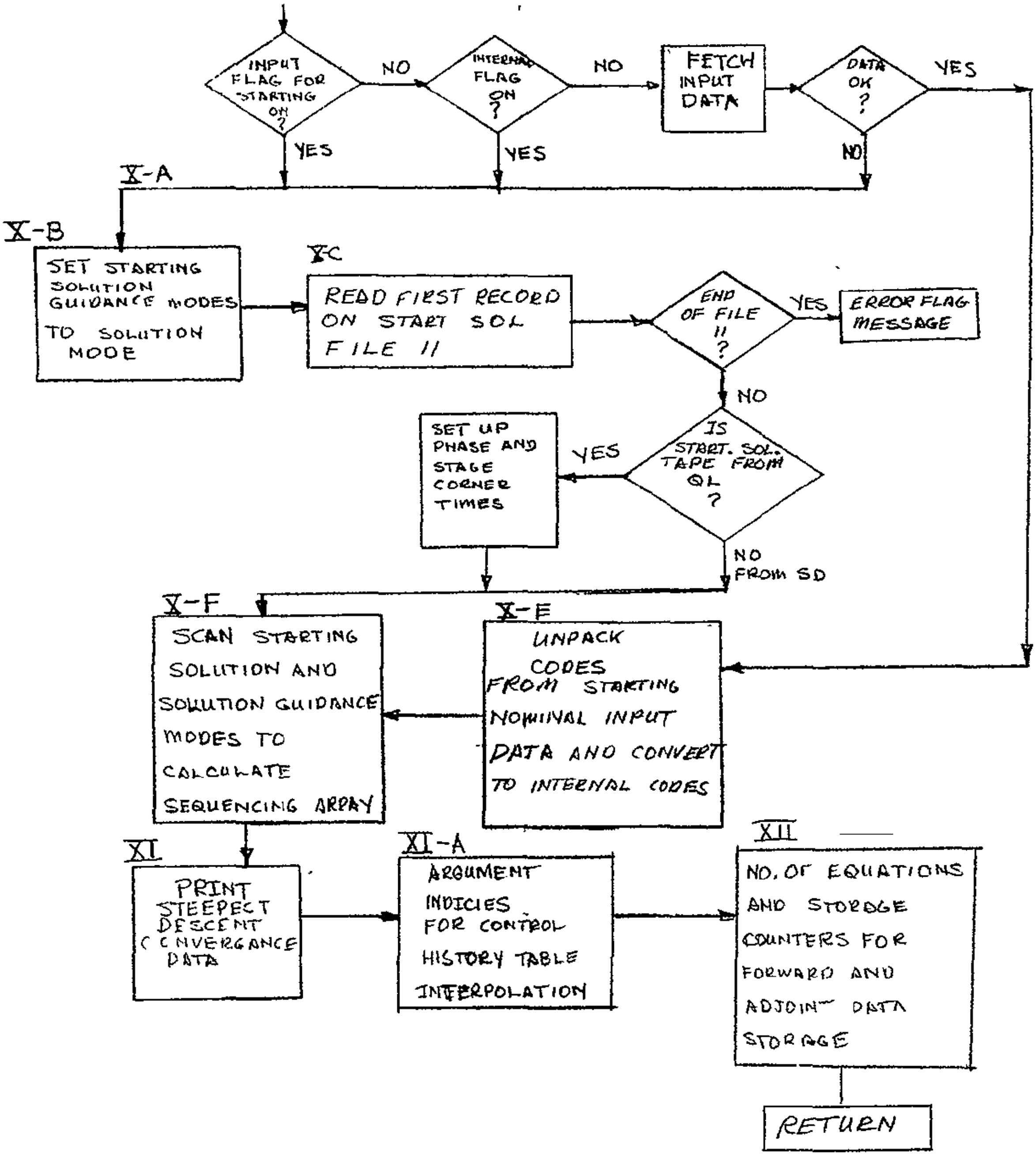
SDINP (CONTINUED)

VII TARGET SCAN



SDINP (CONTINUED)

X STARTING SOLUTION



FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	A	I	Control integral matrix	/GENF	/(109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BST03	I	A
						MTX3A	I	A
						PAY02	I	A
						SDINP	I	A
						TRAN3	I	A
AR		I	Array for storing starting control history tables	/STS	/(112)	SDINP	I	AR
						SDINP	I	III
						TBLK	I	AR
ARCDA	S_{ref}	I	Aerodynamic reference area (FT^2)	/ARCDAT	/(1)	BNTG	I	ARCDA
						EQUA3	I	SREF
						FNTG	I	ARCDA
						FXDAT	I	ARCDA
						FXDAT	O	IARCDA
						GEINP	M	ARCDA
						SDINP	I	ARCDA
						SIZIN	I	ARCDA
						SIZIN	M	SREF
						THRUST	I	SREF
						VT	I	SREF
BNARR		I	Packed boundary condition array.	/SDINP	/(*)	SDINP	I	BNARR
DELP		O	Input or preset nominal parameter adjustment size	/PARAM	/(357)	SDINP	O	DELP
						TOPM	D	DELP
DIP1		M	Phase initial times for nominal trajectory [sd]	/GENF	/(453)	GETIT	I	DIP1
						SDINP	M	DIP1
						TEST	O	DIP1
						TOPM	I	DIP1
DIS1		M	Arc initial times for nominal trajectory [sd]	/GENF	/(473)	GETIT	I	DIS1
						SDINP	M	DIS1
						TEST	O	DIS1
						TOPM	I	DIS1
						TRAN3	I	DIS1
DPAY	$d\Phi$	I	Initial payoff improvement	/STS	/(1)	PAY02	I	DPAY
						SDINP	I	DPAY
						SDINP	O	IST
						SDINP	I	ST
						TEST	I	DPAY
						TOPM	D	IDPAY
G	g	M	Gravitational attraction (FT/SEC**2)	/GENF	/(301)	BL4	I	G
						BL7	I	G
						BL8	I	G
						DER3A	I	G
						EQUA3	M	G
						MODELA	I	G
						MODELB	I	G
						PDY3A	I	G
						SDER3	I	G
						SDINP	M	G

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
GR	g_r	I	Gravitational acceleration at surface of the earth (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	.I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
IAD		M	Starting address of each control history table	/STS /(	312)	SDINP	M	IAD
						TBLK	I	IAD
IARC		M	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
ICOR		M	Phase sequence array	/XCODES/(	10)	ADJUST	I	ICOR
						FNTG	I	ICOR
						PRMSET	I	ICOR
						SDINP	M	ICOR
ID		I	A four word array containing the basic deck, reference run, case and part case numbers in that order.	/GLOBAL/(	21)	BLICO	I	ID
						FRENCH	I	ID
						GEINP	I	ID
						PADS1	O	ID
						PRINT	I	ID
						SDINP	I	ID
						TOPM	I	ID
						VEHDF	I	ID
IFATAL		M	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADS1	I	IFATAL
						SDINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
III		I	Array for storing starting control history tables	/STS /(	112)	SDINP	I	AR
						SDINP	I	III
						TBLK	I	AR
IMODE		I	Control mode option flag	/ARCDAT/(	8)	FXDAT	M	IMODE
						SDINP	I	IMODE
INP		M	Index of last argument of each control history table	/STS /(	332)	SDINP	M	INP
						TBLK	I	INP

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBR	CODE	VAR
INTB		M	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
						BNTG	I	INTB
						ENVPRM	I	INTB
						FNTG	I	INTB
						SDINP	M	INTB
						TEST	I	INTB
						TRAN3	I	INTB
						TRTOSZ	I	INTB
IPH		M	Phase number	/XCODES/(	143)	ADID3A	I	IPH
						ADJUST	I	IPH
						AST3	I	IPH
						BNTG	M	IPH
						FNTG	M	IPH
						GETIT	I	IPH
						GUI3A	I	IPH
						SDINP	M	IPH
IPOINT		O	Code for each adjustable parameter in steepest descent.	/PARAM /(	1)	ADJUST	I	IPOINT
						PRMSET	I	IPOINT
						SDINP	O	IPOINT
						STAU	I	IPOINT
						TOPM	D	IPOINT
IST	$d\Phi$	O	Initial payoff improvement	/STS /(	1)	PAYO2	I	DPAY
						SDINP	I	DPAY
						SDINP	O	IST
						SDINP	I	ST
						TEST	I	DPAY
						TOPM	D	IDPAY
ISV		O	Saved index of last control history table look-up	/STS /(	352)	SDINP	O	ISV
						TBLK	M	ISV
ITAB		I	A 20 word array containing the number of nonzero state initial conditions specified at the beginning of each subarc	/GLOBAL/(	45)	GEINP	I	ITAB
						SDINP	I	ITAB
						SIZIN	I	ITAB
ITI		O	Optimized arc time flag	/XCODES/(	30)	ADJUST	M	ITI
						FNTG	I	ITI
						SDINP	O	ITI
ITPSO		I	A non zero input value indicates to the steepest descent module that the initial steering angle profiles are stored on logical unit 11.	/GLOBAL/(	93)	SDINP	I	ITPSO
ITQ		M	Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
						ADIC3A	I	ITQ
						ADID3A	I	ITQ
						CON3	I	ITQ
						SDINP	M	ITQ
						STAU	I	ITQ
						TOPM	D	IITQ
JGID		M	Control option flag array	/XCODES/(	32)	BNTG	I	JGID
						FNTG	I	JGID
						SDINP	M	JGID
JPH		M	Phase cut-off option flag	/XCODES/(	72)	BNTG	I	JPH
						FNTG	M	JPH
						SDINP	M	JPH
JST		M	Arc cut-off option flag	/XCODES/(	112)	ADICB3	I	JST
						BNTG	I	JST
						FNTG	I	JST
						SDINP	M	JST
K		M	Storage retrieval flag indicates end of arc, phase, or data.	/XCODES/(	156)	AST3	O	K
						FNTG	O	K
						GETIT	M	K
						MODELA	I	K
						SDINP	M	K
KSOL		M	An internal flag that has the same significance as ITPSO.	/GLOBAL/(	94)	FNTG	I	KSOL
						PADS1	O	KSOL
						SDINP	M	KSOL
KTAB		I	A 20 word array containing the number of state target conditions specified at the end of each subarc.	/GLOBAL/(	25)	GEINP	I	KTAB
						SDINP	I	KTAB
						SIZIN	I	KTAB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
		M	Integration traffic control flag L = 1 means evaluate derivatives = 2 check cut-off = 3 print or cut-off detected	/XCODES/(	177)	BNTG FNTG OUT RKTA3A RKTB3A SDINP	M M I M M M	L L L L L L
LUM		I	Program control flag. LUM = 0 Steepest descent only, LUM = 1. Steepest descent and adjoint transformation stored on tape, LUM = 2 Steepest descent and QL; LUM = 3 QL only.	/GLOBAL/(	6)	AST3 FNTG GEINP PADS1 SDINP TOPM	I I I M I M	LUM LUM LUM LUM LUM LUM
MIXA		I	Maximum number of words in trajectory data buffer = 990.	/RETREV/(	12)	AGETB3 AST3 SDINP TOPM	I I I D	MIXA MIXA MIXA MIXA
MIXB		I	Maximum number of words in adjoint data buffer = 3000.	/RETREV/(	13)	BGET3 BSTO3 SDINP TOPM	I I I D	MIXB MIXB MIXB MIXB
MNGA		M	Control history curve number	/STS /(	32)	GUI3A SDINP SDINP	M M I	MNGA MNGA TAL
MNGP		M	Control history curve number	/STS /(	72)	GUI3A SDINP SDINP	M M I	MNGP MNGP WTPI
MXA		O	Index of last stored word in full buffer of forward trajectory data.	/RETREV/(	14)	AGETB3 AST3 SDINP	I I O	MXA MXA MXA
MXB		O	Index of last stored word in full buffer of adjoint data.	/RETREV/(	15)	BGET3 BSTO3 SDINP	I I O	MXB MXB MXB
NARC	N_3	I	Number of subarcs in the problem.	/GLOBAL/(	18)	FNTG GEINP PROPIN SDINP SIZIN	I M I I I	NARC NARC NARC NARC NARC
NBRAN	N_1	O	Number of the last subarc on the stem of a branch problem. If the problem is not a branch problem, then NBRAN = 0.	/GLOBAL/(	19)	SDINP	O	NBRAN
NCNST	n	M	Number of problem constraints	/XCODES/(	132)	BGET3 BSTO3 CON3 PAYO2 SDINP SUMS TEST TOPM TRAN3	I I I I M I I I I	NCNST NCNST NCNST NCNST NCNST NCNST NCNST NCNST NCNST
NEQ		M	Number of integrated states	/XCODES/(	162)	ADICB3 ADIC3A ADID3A AGETB3 AST3 BGET3 BSTO3 MTX3A OUT REU3 SDER3 SDINP TOPM TRAN3 YREF3	I I I I I I I I I I I M I I I	NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ NEQ

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NEQF		O	Number of equations to be integrated on forward trajectory	/XCODES/(	185)	REU3	I	NEQF
						RKTA3A	I	NN
						SDINP	O	NEQF
						STAU	I	NEQF
						TOPM	O	NEQF
						TRAN3	O	NEQF
NFARC	N_2	O	Number of the last subarc on the first branch. If the problem is not a branch problem, then NFARC = NARC.	/GLOBAL/(	20)	SDINP	O	NFARC
NICNB		M	Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
						ADIC3A	I	NICNB
						BNTG	I	NICNB
						REU3	I	NICNB
						SDINP	M	NICNB
						TEST	I	NICNB
						TRAN3	I	NICNB
NITER		I	Maximum number of steepest descent iterations	/STS /(	31)	SDINP	I	NITER
						TEST	I	NITER
NPARA		M	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	O	NPARA
NPH		M	Number of phases in trajectory	/XCODES/(	164)	BNTG	I	NPH
						FNTG	O	NPH
						PRMSET	I	NPH
						SDINP	M	NPH
						TEST	I	NPH
						TOPM	I	NPH
NPTA		M	Number of words stored at each trajectory time point.	/RETREV/(	16)	SDINP	M	NPTA
NPTB		M	Number of words stored at each adjoint solution time point.	/RETREV/(	17)	BGET3	I	NPTB
						SDINP	M	NPTB
NS		I	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST
NSAB		M	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB
NSB		M	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NST		M	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST
OMG	Ω_J	M	Array of arc cut off values [sd]	/GENF /(	1)	ADJUST	M	OMG
						FNTG	I	OMG
						PRMSET	M	OMG
						PROPB	I	OMG
						SDINP	M	OMG
						STP3	I	OMG
						TOPM	O	IOMG
OMGP		M	Array of phase cut off values [sd]	/GENF /(	21)	ADJUST	O	OMGP
						FNTG	M	OMGP
						PRMSET	O	OMGP
						SDINP	M	OMGP
OMGZ	ω	I	Earth rotation rate (RAD/SEC)	/GLOBAL/(	3)	ADID3A	I	OMGZ
						CRASH	I	OMEGA
						DER3A	I	OMGZ
						EQUA3	I	OMGZ
						GEINP	I	OMGZ
						MODELA	I	OMGZ
						MODELB	I	OMGZ
						PDBC	I	OMGZ
						PDY3A	I	OMGZ
						SDINP	I	OMGZ
						TOPM	I	OMGZ
PMIN		I	Minimum payoff improvement	/STS /(	2)	PAY02	I	PMIN
						SDINP	I	PMIN
						TEST	I	PMIN
PSIRF	ψ_r	M	Reference azimuth. (DEG)	/GLOBAL/(	68)	GEINP	I	PSIRF
						REU3	I	PSIRF
						SDINP	M	PSIRF
RAD		I	Radian to angle conversion, 57.29577951	/DATA /(	2)	BEROCO	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADS1	O	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD
RDI		I	Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	O	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI
SIG		M	Payoff sign SIG < 0 Payoff to be minimized, SIG > 0 Payoff to be maximized	/GLOBAL/(	65)	PAY02	I	SIG
						SDINP	M	SIG
						TEST	I	SIG
						TRAN3	I	SIG

654

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ST	$d\Phi$	I	Initial payoff improvement		/STS /(	1)	PAYO2	I	DPAY
							SDINP	I	DPAY
							SDINP	O	IST
							SDINP	I	ST
							TEST	I	DPAY
							TOPM	O	IDPAY
SVAR	$y\|_{t=0}$	M	Array of state values at initial problem time [sd]		/GENF /(	79)	ADJUST	O	SVAR
							BNTG	I	SVAR
							FNTG	I	SVAR
							PRMSET	M	SVAR
							REU3	I	SVAR
							SDINP	M	SVAR
							TEST	I	SVAR
							TOPM	I	SVAR
							TRTOSZ	I	SVAR
TAL		I	Control history curve number		/STS /(	32)	GUI3A	M	MNGA
							SDINP	M	MNGA
							SDINP	I	TAL
TARG		I	Target condition part of BNARR.		/SDINP /(*	)	SDINP	I	TARG
TINIT	t_0	O	Trajectory start time.	(SEC)	/GLOBAL/(	7)	SDINP	O	TINIT
TOL		M	Tolerence on constraint misses [sd]		/GENF /(	70)	SDINP	M	TOL
							SUMS	I	TOL
							TEST	I	TOL
TPH1		O	Phase end times for nominal trajectory		/GENF /(	413)	BNTG	I	TPH1
							GETIT	I	TPH1
							SDINP	O	TPH1
							TEST	O	TPH1
							TOPM	I	TPH1
TST1		O	Arc end times for nominal trajectory		/GENF /(	433)	BNTG	I	TST1
							GETIT	I	TST1
							PROPIN	I	TST1
							SDINP	O	TST1
							TEST	O	TST1
							TOPM	I	TST1
							TRAN3	I	TST1
							TRTOSZ	I	TST1
VARQ	$\langle VARQ \rangle$	M	Desired constraint values [sd]		/GENF /(	61)	CON3	I	VARQ
							SDINP	M	VARQ
							TEST	I	VARQ
WDC		M	Array of drop weight per arc[sd]	(LBS)	/GENF /(	89)	BNTG	I	WDC
							REU3	I	WDC
							SDINP	M	WDC
WORK		M	Working array, contains TOPEN1, TOPEN2, and PHIWT		/STS /(	3)	ADEQ3A	I	WORK
							FNTG	I	WORK
							MODELB	O	WORK
							MTX3A	I	WORK
							SDINP	M	WORK
							TEST	M	WORK
WTP	$[Y]$	O	Adjustable parameter diagonal weighting matrix order according to IPOINT		/PARAM /(	132)	MTX3A	I	WTP
							PAYO2	I	WTP
							SDINP	O	WTP
							TOPM	D	WTP
WTPD		M	Input or preset adjustable parameter weighting factors according to type of parameter		/PARAM /(	123)	SDINP	M	WTPD
							TOPM	D	WTPD
WTPI		I	Control history curve number		/STS /(	72)	GUI3A	M	MNGP
							SDINP	M	MNGP
							SDINP	I	WTPI
XLAMRF	ρ_r	M	Reference latitude	(DEG)	/GLOBAL/(	4)	CRASH	M	RHOO
							GEINP	I	XLAMRF
							REU3	I	XLAMRF
							SDINP	M	XLAMRF
YMURF	μ_r	M	Reference longitude	(DEG)	/GLOBAL/(	5)	CRASH	M	UMUO
							GEINP	I	YMURF
							REU3	I	YMURF
							SDINP	M	YMURF

```
      PROGRAM SDINP                                                     SDINP
C                                                                       COMN
C        SCANS  AND INTERPRETS  INPUT DATA                              COMN
      DIMENSION LABEL(10),LABL2(35), LABL3(2)                           SDINP
      DATA  LABEL /   10H      TIME=, 10H VELOCITY=, 10HPATH ANGL=,     SDINP
     110H ALTITUDE=, 10H SLG.MASS=, 10H AZIMUTH= ,10H LATITUDE= ,       SDINP
     210H LNGITUDE=,10H HEAD LOD=,  10H SQUIGLE2=/                      SDINP
      DATA  LABL2 /                                                     SDINP
     1 10HELPSD.TIME, 10H  VELOCITY, 10HPATH ANGLE, 10H  ALTITUDE,      SDINP
     210H       MASS , 10H   AZIMUTH , 10H  LATITUDE  ,10H LONGITUDE  , SDINP
     310HHEAT LOAD ,10HSQUIGGLE 2,  10HINER VELOC,10HIN.PTH ANG,        SDINP
     410HIN.AZIMUTH,10HIN LONGTUD,  10HSMLAT RCTM,10HECCNTRCITY,        SDINP
     510HINCLNATION,10HARG PERGEE,  10HLNG ASC ND,10HSMIMAJAXIS,        SDINP
     610HAPOGEE RAD,10HPERGEE RAD,  10HTRU ANOMLY,10H CAP X     ,       SDINP
     710H CAP  Y    ,10HASSYMPTOTE,  10H    ENERGY,10H MOMENTUM ,       SDINP
     810H DWN RANGE,10H CRS RANGE,  10H TOT RANGE, 10H DYN PRESS,       SDINP
     *10HHEAT RATE   ,10H REYN NO      ,10H PAYLOAD      /              OS
      DATA LABL3 /    10HMINIMIZED , 10HMAXIMIZED  /                    SDINP
      DIMENSION  SRIF(40)                                               SDINP
      EQUIVALENCE ( SRIF(12),GMIX),(SRIF(13),XLMIX),(SRIF(14),TEMIX),   SDINP
     1  (SRIF(8 ),ICODE)                                                SDINP
      EQUIVALENCE (SRIF,TARG(50))                                       SDINP
      EQUIVALENCE(NS,NST)                                               SDINP
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                           RETREV
      COMMON/RETREV/                                                    RETREV
     *NBUFA(2)  ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA     , RETREV
     *MIXB      ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB    , RETREV
     *,NBUFB    ,IBUFB                                                  RETREV
      COMMON/STS/                                                       STS
     *DPAY      ,PMIN      ,WORK (20),NWDS      ,IPC    (7),NITER     , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)           STS
      LOGICAL IFATAL                                                    SDINP
      INTEGER DEVTYP                                                    SDINP
      COMMON /LASTAB / LSTWD                                            SDINP
      DIMENSION ST(1),IST(1)                                            SDINP
      EQUIVALENCE (DPAY,ST)       ,(DPAY,IST)                           SDINP
      DIMENSION GUARR(1),GUIC(1),III(1)                                 SDINP
      EQUIVALENCE(BNARR,GUARR),(BNARR,GUIC),(AR,III)                    SDINP
      DIMENSION  INPERR(2),ITL(2)                                       SDINP
      DATA INPERR/ 20HG FATAL INPUT ERROR /                             SDINP
      COMMON/GENF/                                                      GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R         ,RE        ,MACH      ,PA        ,RD        ,CS       , GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR        ,DV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT     ,CT        ,CALPHA    ,COE       ,DELTAE    ,SID      , GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED    ,IRATED    ,                                            GENF
     *P1        ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1Q      ,XK2Q      ,XK3Q     , GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV        ,PG        ,PP        ,PR        ,PQ        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,               MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)           ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
```

659

```
      *I2OP      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      ,  XCODES
      *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   ,  XCODES
      *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS       ,  XCODES
      *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE    ,  XCODES
      *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        ,  XCODES
      *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    ,  XCODES
      *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        ,  XCODES
      *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB     ,  XCODES
      *NCTIN     ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
       COMMON/DATA/                                                         DATA
      *PI        ,RAD       ,RDI       ,SC        ,UMF       ,TMPF     ,  DATA
      *FTNM      ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4        DATA
       COMMON/GLOBAL/                                                       GLOBAL
      *GR        ,ER        ,OMGZ       ,XLAMRF    ,YMURF     ,LUM         GLOBAL
      *,JJOP(10) ,IFATAL    ,NARC        ,NBRAN    ,NFARC     ,ID(4)       GLOBAL
      *,KTAB(20),ITAB(20)  ,SIG     ,MAXTAB                                 GLOBAL
      *,GM       ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)         GLOBAL
      *,ITPSO ,KSOL     ,KGLOBL(8)                                          RETAP
       COMMON/ARCDAT/                                                       ARCDAT
      *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI     ,  ARCDAT
      *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX     ,  ARCDAT
      *XLMAX     ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA     ,  ARCDAT
      *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG     ,  ARCDAT
      *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB     ,  ARCDAT
      *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT       ,  ARCDAT
      *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                    ARCDAT
      *  ,FRATE  ,  ARCD(9)                                                 RETAP
       DIMENSION  ARCDA(40)                                                 ARCDAT
       EQUIVALENCE(SREF,ARCDA)                                              ARCDAT
       DIMENSION BNARR(400),TARG(100)                                       SDINP
       COMMON/PARAM/                                                        PARAM
      *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),              PARAM
      *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                          PARAM
      *,DELP(9)                                                             PARAM
       EQUIVALENCE(TINIT,JJOP(1))                                           SDINP
       EQUIVALENCE(BNARR(301),TARG(1))                                      SDINP
       EQUIVALENCE (TAL,MNGA),(WTPI,MNGP),(DLPI,MNGP(20))                   RETAP
       DIMENSION  TAL(40),WTPI(9),DLPI(9)                                   RETAP
C                                                                           COMN
C        I           FLAGS   AND COUNTERS  SET                              COMN
    10 CONTINUE                                                             SDINP
       NICNB=0                                                              SDINP
       NSAB=0                                                               SDINP
       NSB=0                                                                SDINP
       INTB=0                                                               SDINP
       NCNST=0                                                              SDINP
       K= ITAB(1)                                                           SDINP
       ITI=0                                                                SDINP
       NPARA =0                                                             SDINP
       NST=NARC                                                             SDINP
       IOP=0                                                                SDINP
C                                                                           COMN
C        I-A      IF QL  ONLY (LUM=3)  SKIP  SD  INPUT (PC=5)               COMN
       IF(LUM.EQ.3) GO TO 19                                                RETAP
C                                                                           COMN
C        II       STEEPEST DESCENT  CONVERGANCE  AND  CONTROL  DATA         COMN
       MN=5                                                                 RETAP
       CALL  TABIN(DUMMY,1,ST,111,AR,200,ID,MN,0,IEOD)                      RETAP
       LSTW5=LSTWD                                                          NOS
       IF(IEOD.EQ.0) GO TO 11                                               RETAP
       WRITE(6,700) MN                                                      RETAP
       IFATAL =.TRUE.                                                       RETAP
       GO TO 19                                                             RETAP
C        II-A     **                                                        COMN
C           CHECK  FOR  INPUT CONSTRAINT  TOLERANCES  IN  ROUTINE  TOLPSI   RETAP
    11 DUMMY = SETTOL(TAL)                                                  RETAP
C        II-B                                                               COMN
C            CHECK  AND  SET  PARAMETER  WEIGHTING                          RETAP
    12 DO 14 I=1,9                                                          RETAP
       IF(WTPI(I).GT.0.) WTPD(I)=WTPI(I)                                    RETAP
       IF(DLPI(I).GT.0.) DELP(I)=DLPI(I)                                    RETAP
    14  CONTINUE                                                            RETAP
C                                                                           COMN
C        III      BOUNDARY  CONDITION  TABLE (CONSTRUCTED IN  BNDRYC)       COMN
```

20 OCT 72 6.01-46

657

```
      19 IJ=0                                                          APR72
         CALL READMS(9,BNARR,400,21)                                   SDINP
         WRITE(6,20)                                                   SDINP
      20 FORMAT(1H1,40X,36HFIRST-SCAN INPUT BOUNDARY CONDITIONS //)    SDINP
C                                                                      COMN
C         III-A  ERROR  IF NO  INITIAL CONDITIONS FOR  ARC 1           COMN
         IF(K GT 0) GO TO 40                                           SDINP
         IFATAL= TRUE.                                                 SDINP
         WRITE (6,30) INPERR                                           SDINP
      30 FORMAT(2A10/ 33H  NO INITIAL CONDITIONS FOR ARC 1   )         SDINP
         GO TO 150                                                     SDINP
C                                                                      COMN
C         III-B   FIRST ARC  INITIAL CONDITION SET-UP                  COMN
      40 CONTINUE                                                      SDINP
         DO 140 I=1,K                                                  SDINP
         IK = BNARR(IJ+2) + 5                                          SDINP
         II = BNARR(IJ+1)+.5                                           SDINP
         GO TO (50,60,70,60,80,70,70,70,90,60,60), IK                  SDINP
      50 IF(II EQ 4) GO TO 100                                         SDINP
         JST(1)= 2                                                     SDINP
         OMG(1)= BNARR(IJ+3)                                           SDINP
         GO TO 110                                                     SDINP
      60 SVAR(IK)= BNARR(IJ+3)                                         SDINP
         GO TO 110                                                     SDINP
      70 SVAR(IK)= BNARR(IJ+3)* RDI                                    SDINP
         GO TO 110                                                     SDINP
      80 SVAR(IK)= BNARR(IJ+3)/GR                                      SDINP
         GO TO 110                                                     SDINP
      90 SVAR(1)= BNARR(IJ+3)                                          SDINP
         GO  TO 110                                                    SDINP
     100 JST(1)= 999                                                   SDINP
     110 IJ= IJ+3                                                      SDINP
C         III-C   TEST  FOR   OPTIMIZED  INITIAL  STATE  AND IF SO     COMN
C                 SET   PARAMETER  FLAGS                               COMN
         IF(II.NE.2) GO TO  140                                        SDINP
         WRITE(6,130) LABEL(IK)                                        SDINP
         NPARA = NPARA +1                                              SDINP
         IF(IK EQ 1) GO TO 120                                         SDINP
         IPOINT(NPARA)= 20 +IK                                         SDINP
         WTP(NPARA)=WTPD(IK)                                           SDINP
         GO TO 140                                                     SDINP
     120 IPOINT(NPARA) = 1                                             SDINP
         WTP(NPARA) = WTPD(1)                                          SDINP
     130 FORMAT(10X, 8HINITIAL A9,16H TO BE OPTIMIZED)                 SDINP
     140 CONTINUE                                                      SDINP
C         III-D                                                        COMN
C           TEST FOR REFERENCE  POSITIONS                              P014
         IF(ABS(PSIRF) GT 0 ) GO TO 141                                P014
         IF(SIGN(1 ,PSIRF) LT 0 ) PSIRF=SVAR(6)*RAD                    P014
     141 IF(ABS(XLAMRF) GT 0 ) GO TO 142                               P014
         IF(SIGN(1 ,XLAMRF) LT 0 ) XLAMRF =SVAR(7)*RAD                 P014
     142 IF(ABS(YMURF) GT 0 ) GO TO 143                                P014
         IF(SIGN(1.,YMURF) LT 0 ) YMURF=SVAR(8)*RAD                    P014
     143 CONTINUE                                                      P014
         OMEGA=OMGZ                                                    P014
         OMEGA2=OMGZ*OMGZ                                              P014
C                                                                      COMN
C         III-E     WRITE OUT STATE  ICS   AND IF APPLICABLE           COMN
C                   ARC-TIME  CUT-OFF                                  COMN
         JJJ=1                                                         SDINP
         KM1=K-1                                                       SDINP
         WRITE(6,970)(( LABEL(I),SVAR(I)),I=1,KM1)                     SDINP
         I=1                                                           SDINP
         IF(JST(1).EQ 2) WRITE(6,980) I,LABEL(1),OMG(1)                SDINP
C                                                                      COMN
C          IV       FIRST   ARC  TARGETS                               COMN
     150 IK=0                                                          SDINP
         IF(KTAB(1).EQ 0) GO TO 230                                    SDINP
```

20 OCT 72 6.01-46

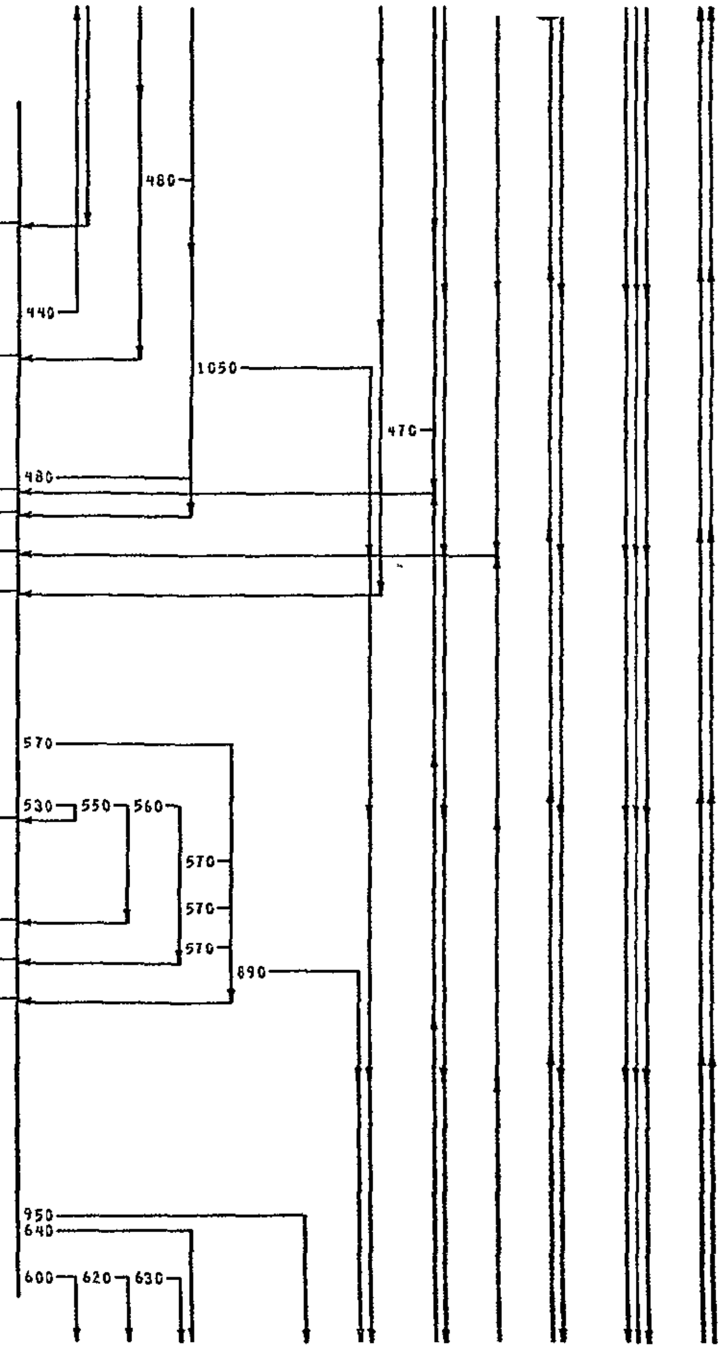
```
C       IV-A    IF  NO  TARGETS  GO  TO  NEXT ARC                    COMN
        MN=KTAB(1)                                                   SDINP
        DO 220 I=1,MN                                                SDINP
C        IV-B     UNPACK OPTION CODE, LC AND VARIABLE CODE,ILV       COMN
        LQ= ICOD(TARG(IK+1))                                         SDINP
        LV= IVOD(TARG(IK+1))                                         SDINP
        ILV = IABS( LV)                                              SDINP
        LC=IABS(LQ)                                                  SDINP
C        IV-C    TEST  FOR  CUT-OFF  CODE (LC=1)                     COMN
        IF(LC.NE.1) GO TO 170                                        SDINP
C              ERROR  IF  ARC-TIME CUT-OFF ALREADY SET               COMN
        IF(JST(1) NE.999)GO TO1110                                   SDINP
        JST(1)= MOMG(LV)                                             SDINP
C         IV-D     SET  INTERNAL  CUT-OFF  CODE  (ERROR IF INVALID CODE)   COMN
C             ALSO  SET  VALUE OF  CUT-OFF                           COMN
        IF(JST(1) EQ 99) GO TO 210                                   SDINP
        OMG(1)= SOMG(ILV,TARG(IK+2))                                 SDINP
    160 IK=IK+2                                                      SDINP
        ILK=LOMG(JST(1))                                             SDINP
C         IV-E     PRINT   STOPPING  CONDITION                       COMN
        WRITE (6,980) JJJ,LABL2(ILK),OMG(1)                          SDINP
        GO TO 220                                                    SDINP
    170 CONTINUE                                                     SDINP
C         IV-F   NOT A  CUT-OFF  ,IF  NARC GT 1  MUST BE AN INTERMEDIATE  COMN
C          CONSTRAINT   GO  TO   IV-H                                COMN
        IF(NS NE 1) GO TO 200                                        SDINP
C       THIS IS ONLY A ONE ARC PROBLEM                               SDINP
C          SET  INTERNAL  CONSTRAINT  CODES  AND  VALUES             COMN
C          AND  COUNT  NO OF CONSTRAINTS,                            COMN
        IF(LC EQ 2) GO TO 190                                        SDINP
        NCNST =NCNST +1                                              SDINP
    180 ITQ(NCNST)= MPSI( ILV)                                       SDINP
        ITQI  = ITQ(NCNST)                                           SDINP
        IF(ITQI EQ.99) GO TO 210                                     SDINP
        VARQ(NCNST)= SPSI(   ILV    ,TARG(IK+2))                     SDINP
        TOL(NCNST) = TOLPSI(ILV)                                     SDINP
        IF(ITQI.GE 10) ITQI= ITQI-1                                  SDINP
        WRITE(6,990) JJJ,LABL2(ITQI),VARQ(NCNST),TOL(NCNST)          SDINP
        IK=IK +2                                                     SDINP
        GO TO 220                                                    SDINP
    190 CONTINUE                                                     SDINP
C        IV- G     ERROR IF MULTIPLY DEFINED PAYOFF,ELSE             COMN
C              SET  PAYOFF  VALUES  AND  CODES AS WELL AS  SIG (+MAX,-MIN  COMN
        IF(IOP.NE 0) GO TO 1040                                      SDINP
        IOP =  MPSI(ILV)                                             SDINP
        IF(IOP.EQ 99)GO TO 210                                       SDINP
        SIG=ISIGN(1,LQ)                                              SDINP
        VALOP= SPSI( ILV, TARG(IK+2))                                SDINP
        IK=IK+2                                                      SDINP
        GO TO 220                                                    SDINP
C          **  IV-H                                                  COMN
C     A BRANCHING  OR INTERMEDIATE  CONSTRAINT  PROBLEM              SDINP
    200 CONTINUE                                                     SDINP
        NCNST=NCNST+1                                                SDINP
        INTB =1                                                      SDINP
        NSB =1                                                       SDINP
        NICNB =NICNB +1                                              SDINP
        GO TO 180                                                    SDINP
C           IV - I      ERROR  FLAG  SET                             COMN
    210 IFATAL=.TRUE.                                                SDINP
        IK=IK+2                                                      SDINP
    220 CONTINUE                                                     SDINP
        IF(NS EQ 1) GO TO 510                                        SDINP
C       MORE THAN ONE ARC                                            SDINP
C       BNDRY CONDITIONS FOR  REMAINING ARCS                         SDINP
    230 CONTINUE                                                     SDINP
C             V      INITIAL   CONDITIONS  FOR  REMAINING  ARCS      COMN
        DO 500 I=2,NS                                                SDINP
        K=ITAB(I)                                                    SDINP
        MN=KTAB(I)                                                   SDINP
        IF(K.GT 0) GO TO 240                                         SDINP
```

665

```
C        V-A    ERROR IF NONE (ARC TIME CODE MUST AT LEAST BE DEFINED) COMN
      GO TO 1110                                                       SDINP
  240 DO 400 L=1,K                                                     SDINP
C         V-B   SET VARIABLE AND OPTION  CODES                         COMN
      IVD= BNARR(IJ+2) + 5                                             SDINP
      ICO= BNARR(IJ+1) + 5                                             SDINP
C              IGNORE  CONTINUITY  CODE  SETS                          COMN
      IF(ICD EQ.0) GO TO 390                                           SDINP
C        V-C  NOT ARC-TIME CODE  GO TO  V-H                            COMN
      IF(IVD NE 1) GO TO  270                                          SDINP
C        V-D    ARC  TIME  UNKNOWN  GO TO V-G                          COMN
      IF(ICD.EQ 4) GO TO  260                                          SDINP
C        V-E     SET  ARC-CUT-OFF TIME   AND PRINT  STOPPING CONDITION  COMN
      JST(I) =2                                                        SDINP
      OMG(I)= BNARR(IJ+3)                                              SDINP
      WRITE(6,980) I,LABEL(1),OMG(I)                                   SDINP
C        V-F     IF  ARC-TIME OPTIMIZED   SET  PARAMETER DATA AND  PRINT COMN
      IF(ICD NE.2) GO TO 390                                           SDINP
      WRITE(6,250)                                                     SDINP
      NPARA = NPARA +1                                                 SDINP
      IPOINT(NPARA)= I                                                 SDINP
      WTP(NPARA)= WTPD(1)                                              SDINP
  250 FORMAT(10X,31HTHIS ARC TIME WILL BE OPTIMIZED)                   SDINP
      GO TO  390                                                       SDINP
C        V-G     SET FLAG  TO INDICATE  CUT-OFF IS A  TARGET           COMN
  260 JST(I)=999                                                       SDINP
      GO TO 390                                                        SDINP
C        V-H     INITIAL  CONDITION  OTHER  THAN  ARC TIME             COMN
C                TEST  FOR  BRANCH  INDICATOR ,YES  GO TO VI-O         COMN
  270 IF(ICO.GT 10) GO TO 350                                          SDINP
C          V-I     TEST  FOR  WEIGHT  CODE   NO, GO TO  V- R           COMN
      IF(IVD.NE.5)  GO TO 320                                          SDINP
C        V-J     IF BRANCHING  AND  AT FIRST ARC OF BRANCH 2 GO TO VI-B COMN
      IF(INTB EQ 2 AND I EQ NSB+NSAB+1)  GO TO 330                     SDINP
C        V-K    IF DISCONT  CODE (=5)  GO TO  V-O                      COMN
      IF(ICD EQ 5) GO TO 290                                           SDINP
C      V-L   IF (WTDROP) CODE  GO TO  V-P                              COMN
      IF(ICD EQ 6) GO TO  295                                          PHISZ
C      V-M   IF  PREVIOUS ARC  DID NOT CUT-OFF ON WEIGHT  GO TO V-Q    COMN
      IF(IABS(JST(I-1)) NE 6) GO TO 300                                SDINP
C        V-N    COMPUTE  WEIGHT DROPPED  AND  PRINT                    COMN
      WDC(I-1)=  OMG(I-1)*GR - BNARR(IJ+3)                             SDINP
  280 WRITE(6,1000) I, WDC(I-1)                                        SDINP
      GO  TO  390                                                      SDINP
  290 WDC(I-1)= BNARR(IJ+3)                                            SDINP
C      V-O     SET  WEIGHT DROPPED  TO INPUT  VALUE  AND  PRINT        COMN
      GO TO 280                                                        SDINP
C      V-P  **                                                         COMN
C           OPTIMAL  STAGING  WT  DROP                                 PHISZ
  295 WDC(I-1) = -99999999.                                            PHISZ
      IM1=I-1                                                          PHISZ
      WRITE(6,1005) IM1                                                PHISZ
 1005 FORMAT (//,3X,21HWEIGHT DROPPED IN ARC,I3,30H CALCULATED IN SUBR  PHISZ
     *(WTDRP)  )                                                       PHISZ
      GO  TO 390                                                       PHISZ
C      V-Q     WEIGHT INITIALIZATION  AND  PRINT                       COMN
  300 WDC(I-1)=-BNARR(IJ+3)                                            SDINP
  301 CONTINUE                                                         OS
      WD=-WDC(I-1)                                                     SDINP
      WRITE(6,310) I,WD                                                SDINP
  310 FORMAT(//                                                        SDINP
     13X, 33HKNOWN WEIGHT AT BEGINNING OF ARC I3,2H =E17.8,5H LBS  )   SDINP
      GO TO 390                                                        SDINP
C      V-R     NON- WEIGHT  ICS                                        COMN
C              IF  KNOWN CODE  OK  OTHERWISE  ERROR                    COMN
C              IF  BRANCH  2   GO  TO    VI-C                          COMN
C              IF  PRIOR  CUT-OFF  AND  IC NOT COMPATIBLE - ERROR      COMN
  320 IF(ICD NE 1) GO TO 1070                                          SDINP
C TEST FOR  PRIOR STAGING                                              SDINP
      IF(INTB EQ 2 AND I EQ NSB+NSAB+1 ) GO TO 340                     SDINP
      IF(MOMG(IVD) NE.IABS(JST(I-1))) GO TO 1080                       SDINP
```

```
      GO  TO  390                                                    SDINP
C       VI      BRANCH  LOGIC                                        COMN
C                                                                    OS
C        VI-A WT  DROP ,CONTINUITY, OR SUM OF PARTS                  OS
  330 IF(ICO NE 5) GO TO 332                                         OS
      IF(IABS(JST(NSB)) NE 6) GO TO 290                              OS
      WDC(I-1) =-(OMG(NSB)-BNARR(IJ+3))                              OS
      GO TO 301                                                      OS
  332 IF(ICO EQ 1) GO TO 300                                         OS
      IF(ICO.NE 7) GO TO 1090                                        OS
C                                                                    OS
C    ASSUMES WT  = SUM OF PARTS                                      OS
      WDC(I-1)=0                                                     OS
      GO TO 390                                                      OS
C       VI-C      IF PRIOR ARC  CUT OFF NOT WEIGHT - ERROR           COMN
  340 IF(IVD NE IABS(JST(NSB)))  GO TO 1080                          SDINP
      GO  TO 390                                                     SDINP
C        VI-D      ICD   TOO  LARGE   -ERROR  ELSE  SET  BRANCHING FLAGS COMN
C                  AND  KEEP COUN  OF  NUMBER OF  CONSTRAINTS  AT    COMN
C                  TERMINUS OF FIRST BRANCH                          COMN
  350 IF(ICD GT 29) GO TO 1090                                       SDINP
      LBP =  ICD -10                                                 SDINP
      IF(INTB EQ 1) GO TO 360                                        SDINP
      IF(NSB NE 0) GO TO 370                                         SDINP
  360 NSB = LBP                                                      SDINP
      IF(NSB GT I-1) GO TO 1090                                      SDINP
      INTB =2                                                        SDINP
      NSAB = I- NSB -1                                               SDINP
      NICNB =NCNST                                                   SDINP
      WRITE(6,1010)  NSB,NSAB,NICNB                                  SDINP
      GO TO  390                                                     SDINP
  370 IF(LBP EQ.NSB)GO TO 390                                        SDINP
      GO TO 1090                                                     SDINP
  380 IFATAL= TRUE                                                   SDINP
  390 IJ=IJ+3                                                        SDINP
  400 CONTINUE                                                       SDINP
C                                                                    COMN
C         VII    TARGETS  SCAN                                       COMN
C                  CHECK FOR  NO  CUT-OFF  -ERROR ,ELSE GO TO VIII   COMN
  410 IF(MN GT 0) GO TO 420                                          SDINP
      IF(JST(I) EQ.999) GO TO 1100                                   SDINP
      GO TO  500                                                     SDINP
  420 DO 490  L=1,MN                                                 SDINP
C         VII-A   UNPACK TARGET  CODES                               COMN
      LQ= ICOD(TARG(IK+1))                                           SDINP
      LV= IVOD(TARG(IK+1))                                           SDINP
      LC=IABS(LQ)                                                    SDINP
      ILV= IABS(LV)                                                  SDINP
C         VII-B   IF NOT-CUT-OFF CODE  GO TO   VII-E                 COMN
      IF(LC NE 1) GO TO 430                                          SDINP
C         VII-C   IF   CUT-OFF MULTIPLY DEFINED  -ERROR              COMN
      IF(JST(I) NE 999) GO TO 1110                                   SDINP
C         VII-D   SET INTERNAL CUT-OFF CODE ,CHECK VALIDITY AND      COMN
C                 PRINT STOPPING  CONDITION                          COMN
      JST(I)= MOMG(LV)                                               SDINP
      IF(JST(I) EQ 99) GO TO 470                                     SDINP
      OMG(I)= SOMG(ILV,TARG(IK+2))                                   SDINP
      ILK=LOMG(JST(I))                                               SDINP
      WRITE(6,980) I,LABL2(ILK),OMG(I)                               SDINP
      GO TO  480                                                     SDINP
C         VII-E    LAST  ARC TEST  IF NOT  GO  TO   VII-H            COMN
  430 IF(NS NE I)  GO TO 450                                         SDINP
C         VII-F     PAYOFF  TEST, YES  GO  TO  VII-I                 COMN
      IF(LC EQ 2) GO TO  460                                         SDINP
C         VII-G                                                      COMN
C     LAST  STAGE  OF  PROBLEM  TERMINAL  CONSTRAINTS                SDINP
      NCNST = NCNST +1                                               SDINP
  440 ITQ(NCNST)= MPSI(ILV)                                          SDINP
      ITQI = ITQ(NCNST)                                              SDINP
      IF(ITQI EQ.99) GO TO 470                                       SDINP
```

```
         VARQ(NCNST)= SPSI( ILV ,TARG(IK+2))                       SDINP
         TOL(NCNST) = TOLPSI( ILV)                                 SDINP
         IF(ITQI.GE.10) ITQI= ITQI-1                               SDINP
         WRITE(6,990) I  ,LABL2(ITQI),VARQ(NCNST),TOL(NCNST)       SDINP
         GO  TO  480                                               SDINP
C           VII-H                                                  COMN
C     BRANCHING OR  INT  CONSTRAINT  PROBLEM                       SDINP
     450 CONTINUE                                                  SDINP
         NCNST=NCNST+1                                             SDINP
         NICNB =NICNB+1                                            SDINP
         INTB =1                                                   SDINP
         NSB = I                                                   SDINP
         GO TO 440                                                 SDINP
C                VII-I    PAYOFF  DEFINITION                       COMN
C                TEST  FOR  MULTIPLE PAYOFF - ERROR                COMN
     460 IF(IOP.NE 0) GO TO 1050                                   SDINP
C                VII-J    SET  TEMP     PAYOFF CODE  CHECK VALIDITY COMN
C                 AND    CALC.  SIG  (+MAX,-MIN)                   COMN
         IOP = MPSI(ILV)                                           SDINP
         IF(IOP EQ.99) GO TO 470                                   SDINP
         VALOP = SPSI(ILV,TARG(IK+2))                              SDINP
         SIG=ISIGN(1,LQ)                                           SDINP
         GOTO 480                                                  SDINP
     470 IFATAL= TRUE                                              SDINP
     480 IK=IK+2                                                   SDINP
     490 CONTINUE                                                  SDINP
     500 CONTINUE                                                  SDINP
C                VII-K    SET  INTERNAL PAYOFF CODES  AND PRINT    COMN
     510 ITQ(NCNST+1) = IOP                                        SDINP
         VARQ(NCNST+1)= VALOP                                      SDINP
         KIK =.5*SIG + 2.                                          SDINP
         IF(IOP GE 10) IOP= IOP-1                                  SDINP
         WRITE(6,1020) NS,LABL2(IOP),LABL3(KIK),VALOP              SDINP
     520 CONTINUE                                                  SDINP
C                VIII   SET  Q-L   BRANCHING  FLAGS  IF (LUM NE 0) COMN
C                 IF  LUM =0   SKIP  SAVE SOLUTION  FILE           COMN
         TINIT=SVAR(1)                                             SDINP
         IF(LUM EQ 0) GO TO 570                                    SDINP
         NBRAN=NSB                                                 SDINP
         NFARC=NSB+NSAB                                            SDINP
         IF(NSB EQ 0) NFARC=NARC                                   SDINP
         GO TO(530,550,560),LUM                                    SDINP
     530 REWIND  11                                                SDINP
         IJ= DEVTYP(11)                                            SDINP
         IF(IJ.EQ.32)GO TO 570                                     SDINP
         WRITE(6,540)                                              SDINP
     540 FORMAT(//,16H ****WARNING****//,29H TAPE 11 HAS NOT BEEN MOUNTED) SDINP
         GO TO 570                                                 SDINP
     550 CONTINUE                                                  SDINP
         GO TO 570                                                 SDINP
     560 IF(IFATAL) GO TO 890                                      SDINP
         RETURN                                                    SDINP
     570 CONTINUE                                                  SDINP
         WRITE(6,580)                                              SDINP
     580 FORMAT(1H1)                                               SDINP
C                IX                                                COMN
C        SOLUTION GUIDANCE  PHASING  EDIT                          SDINP
         WRITE(6,900)                                              SDINP
C                IX-A     **                                       COMN
C       ARC DATA  SCAN                                             SDINP
     590 CONTINUE                                                  SDINP
         L=0                                                       SDINP
C                IX-B   CONVERT  INPUT CONTROL MODES  TO INTERNAL CODES COMN
         DO 680 I=1,NS                                             SDINP
         CALL READMS(9,ARCDA,20,I)                                 SDINP
         IF(IMODE   .GT. 5   ) GO TO 950                           SDINP
         IF(IMODE   .LT.3) GO TO 640                               SDINP
         L=L+1                                                     SDINP
         K=IMODE   - 2                                             SDINP
         GO TO (600,620,630),K                                     SDINP
```



```
      600 JGID(L,2) = 8                                                  SDINP
      610 JPH(L,2)  = JST(I)                                             SDINP
          OMGP(L,2) = OMG(I)                                             SDINP
          GO TO  680                                                     SDINP
      620 JGID(L,2)= 7                                                   SDINP
          GO TO 610                                                      SDINP
      630 JGID(L,2)= 6                                                   SDINP
          GO TO 610                                                      SDINP
      640 CONTINUE                                                       SDINP
      650 JG=10 +IMODE                                                   SDINP
C         IS THIS THE LAST ARC                                           SDINP
      660 IF(I EQ.NS) GO TO 670                                          SDINP
C         IS THIS THE BRANCHPT ARC                                       SDINP
          IF(I.EQ.NSB) GO TO 670                                         SDINP
          IF(I.EQ.NSB+NSAB) GO TO 670                                    SDINP
          CALL READMS(9,SRIF,20,I+1)                                     SDINP
          IF(ICODE       .NE.IMODE    ) GO TO 670                        SDINP
          GO TO 680                                                      SDINP
      670 L=L+1                                                          SDINP
          JGID(L,2) =JG                                                  SDINP
          JPH(L,2) = JST(I)                                              SDINP
          OMGP(L,2)= OMG(I)                                              SDINP
      680 CONTINUE                                                       SDINP
      700 FORMAT(30H  INPUT ERROR IN PART-CASE       ,I3)                LS
      740 WRITE(6,910) (( I,JGID(I,2),JPH(I,2),OMGP(I,2)),I=1,L)         SDINP
          WRITE(6,1030)                                                  SDINP
          MN=4                                                           SDINP
C              X     STARTING  SOLUTION   LOGIC                          COMN
C                   IF  STARTING  SOLUTION ON  FILE  GO TO  X-A          COMN
C                   ELSE  READ IN  DATA  AND GO TO X-D                   COMN
          IF(ITPSO.NE. 0    )  GO TO  741                                RETAP
          IF(KSOL.NE 0) GO TO  741                                       RETAP
          CALL  TABIN (DUMMY,1,RUMMY,1,GUIC,400 ,ID,MN,0,IEOD)           SDINP
          IF(IEOD.EQ 0)GO TO 750                                         SDINP
C           X-A     **                                                   COMN
C             STARTING  SOLUTION ON FILE 11                              RETAP
C   X-B    SET  STARTING GUIDANCE CODES = TO SOLUTION  CODES             COMN
      741 DO  745  I=1,L                                                 RETAP
          JPH(I,1) = JPH(I,2)                                            RETAP
          JGID(I,1)= JGID(I,2)                                           RETAP
          OMGP(I,1)= OMGP(I,2)                                           RETAP
          MNGA(I,1)= MNGA(I,2)                                           RETAP
          MNGP(I,1)= MNGP(I,2)                                           RETAP
      745 CONTINUE                                                       APR72
          NPH=L                                                          PH1SZ
          KSOL =1                                                        RETAP
          REWIND 11                                                      RETAP
C   X-C   READ FIRST RECORD ON  11 AND  CHECK EOF  IF EOF ERROR          COMN
          READ(11)AAAA,NNNN,NNNN,(DIS1(I),I=1,NST),(DIP1(J),J=1,L)       NOS
         *,(TST1(II),II=1,NS),(TPH1(IJ),IJ=1,L)                          NOS
          IF(DIP1(1).GT.0.) GO TO 7045                                   JULY28
          K =1                                                           JULY28
          DO 7041 I=1,NS                                                 JULY28
          TST1(I+1)= DIS1(I)                                             JULY28
          IF(OMG(I).NE.OMGP(K,2)) GO TO 7041                             JULY28
          DIP1(K) = DIS1(I)                                              JULY28
          TPH1(K+1)= DIP1(K)                                             JULY28
          K=K+1                                                          JULY28
     7041 CONTINUE                                                       JULY28
          TPH1(1) = SVAR(1)                                              JULY28
          TST1(1) = SVAR(1)                                              JULY28
     7045 CONTINUE                                                       JULY28
          IF(EOF,11) 742,791                                             RETAP
      742 IFATAL =.TRUE.                                                 RETAP
          WRITE(6,743) INPERR                                            RETAP
      743 FORMAT ( 2A10 /67H  STARTING SOLUTION, FILE 11 , IS  NOT AVAILABLE RETAP
         * FOR  THIS CASE                      )                         RETAP
          GO TO 810                                                      RETAP
C           X-D    **                                                    COMN
C         STARTING  NOMINAL GUIDANCE INPUT  STORED IN GUIC               SDINP
```

```
      750 NPH=0                                                         SDINP
           MN=2                                                         SDINP
C            X-E    UNPACK CODES FROM  STARTING NOMINAL INPUT           COMN
C                   AND  CONVERT TO  INTERNAL  CODES                    COMN
          DO 780 I=1,20                                                 SDINP
           IF(MN+1 GT LSTWD) GO TO 790                                  SDINP
          IPH= GUIC(MN+1)*1.E-6 + 1                                     SDINP
          G= IPH*1.E6                                                   SDINP
          F= GUIC(MN+1) -G                                              SDINP
          IARC= F*1.E-3 +.1                                             SDINP
          G= IARC*1.E3                                                  SDINP
          F= F-G                                                        SDINP
          JG=F  +.1                                                     SDINP
          IC1= GUIC(MN+2)*1 E-6 +.1                                     SDINP
          G=IC1*1 E6                                                    SDINP
          F= GUIC(MN+2) - G                                             SDINP
          IC2 = F + 1                                                   SDINP
          IF(IARC EQ.0) GO TO 760                                       SDINP
          MN=MN+4                                                       SDINP
          JPH(IPH,1) = JST(IARC)                                        SDINP
          OMGP(IPH,1) =OMG(IARC)                                        SDINP
          GO TO 770                                                     SDINP
      760 JP=GUIC(MN+3)    +SIGN( 1,GUIC(MN+3))                         SDINP
          JPH(IPH,1) =MOMG(JP)                                          SDINP
          OMGP(IPH,1)=SOMG(IABS(JP),GUIC(MN+4))                         SDINP
          MN=MN+4                                                       SDINP
      770 JGID(IPH,1)=JG                                                SDINP
          MNGA(IPH,1)=IC1                                               SDINP
          MNGP(IPH,1)=IC2                                               SDINP
          NPH=NPH+1                                                     SDINP
      780 CONTINUE                                                      SDINP
      790 CONTINUE                                                      SDINP
C            X-F     **  PHASE  SEQUENCING  ARRAY                       COMN
C     SCAN NOM AND SOLUTION  GUIDANCE  TO CALC  ICOR                    SDINP
      791 L=1                                                           APR72
          DO 800 I=1,NPH                                                SDINP
          IF(JPH(I,1) NE JPH(L,2)) GO TO 800                            SDINP
          IF(OMGP(I,1) NE OMGP(L,2)) GO TO 800                          SDINP
          ICOR(L) = I                                                   SDINP
          L=L+1                                                         SDINP
      800 CONTINUE                                                      SDINP
          WRITE(6,910) (( I,JGID(I,1),JPH(I,1),OMGP(I,1)),I=1,NPH)      SDINP
          CALL IPR(4HICOR,A,ICOR,10,1)                                  SDINP
C           XI        STEEPEST  DESCENT  CONVERGANCE  DATA  INTERPRETATION  COMN
      810 CONTINUE                                                      RETAP
          DO 840 I=23,31                                                SDINP
      840 IST(I)=ST(I)                                                  SDINP
          WRITE(6,850) DPAY,PMIN,NITER,WORK                             SDINP
      850 FORMAT(//24H  STEEPEST DESCENT INPUT/,35H  INITIAL PAYOFF IMPROVEM SDINP
         1ENT ,DPAY=E15 6,34H ,MINIMUM PAYOFF IMPROVEMENT,PMIN=E15 6/,  SDINP
         227H  MAXIMUM ITERATIONS,NITER=I3/12H  WORK ARRAY/(6E15 6))    SDINP
          N=0                                                           SDINP
          J=III(1)                                                      SDINP
          IF(J.GT.21) GO TO 820                                         SDINP
C           XI-A   DETERMINE  FIRST AND LAST ARGUMENT LOCATION          COMN
C                  IN  CONTROL HISTORY  TABLE  FOR  USE  IN  INTERPOLATION  COMN
C                  ROUTINE (TBLK)                                       COMN
          DO 870 I=2,J                                                  SDINP
          K=I-1                                                         SDINP
          IF(III(I).EQ 0) GO TO 870                                     SDINP
          IAD(K) = III(I) + J                                           SDINP
          ISV(K) = IAD(K)                                               SDINP
          IF(N.EQ.0) GO TO 860                                          SDINP
          INP(N)= IAD(K)-2                                              SDINP
      860 N=K                                                           SDINP
      870 CONTINUE                                                      SDINP
      820 CONTINUE                                                      APR72
          INP(N)=LSTW5+J-2                                              NOS
          CALL IPR(4HIAD ,A,IAD ,J,1)                                   SDINP
          CALL IPR(4HINP ,A,INP,J,1)                                    SDINP
```

670

```
      IF(IFATAL)   GO TO 890                                             SDINP
C       XII    SET NO OF EQUATIONS  TO BE INTEGRATED                    COMN
C              AND  CALCULATE  NO  OF  WORDS  TO BE STORED  IN          COMN
C              TRAJECTORY  AND ADJOINT  BUFFERS                         COMN
      IEQ=3                                                              SDINP
      NEQ= 8                                                             SDINP
      NPTA = NEQ+5                                                       SDINP
  880 MXA=(MIXA / NPTA)* NPTA                                            SDINP
      NPTB =  NEQ*(NCNST+1) + (IEQ-1)*(NCNST+1)  + (NCNST+1)*(NCNST+2)/2 SDINP
      MXB = (MIXB/NPTB) * NPTB                                           SDINP
      NEQF=NEQ                                                           SDINP
      IF(WORK(10).EQ.0.) WORK(10)=1                                      SDINP
      RETURN                                                             SDINP
  890 WRITE(6,940)                                                       SDINP
      RETURN                                                             SDINP
  900 FORMAT(20(1H*),31HSOLUTION TRAJECTORY PHASE INPUT,20(1H*)//)       SDINP
  910 FORMAT(/ 4H ARCI3,14H CONTROL OPT.=I3,11H,STOP OPT.=I3, 4H AT E17. SDINP
     18)                                                                 SDINP
  920 FORMAT( 2A10/, 5H  ARCI3,15H CONTROL MODE =I3,11H IS ILLEGAL)      SDINP
  930 FORMAT( 2A10/,50H  BAD COMBINATION OF INEQUALITY CONSTRAINTS IN AR SDINP
     1CI3)                                                               SDINP
  940 FORMAT(//,72H  INPUT ERRORS HAVE BEEN DETECTED  EXECUTION OF PROBL SDINP
     1EM WILL BE ABORTED    )                                            SDINP
C                                                                        SDINP
  950 WRITE(6,920) INPERR,I,IMODE                                        SDINP
      IFATAL =.TRUE.                                                     SDINP
      GO TO 670                                                          SDINP
C                                                                        SDINP
  960 WRITE(6,930) INPERR,I                                              SDINP
      IFATAL = TRUE                                                      SDINP
      GO TO 670                                                          SDINP
  970 FORMAT( 3X,29HTRAJECTORY INITIAL CONDITIONS/ (4(A10,E17.8)  )      SDINP
  980 FORMAT(//,3X,47HSOLUTION TRAJECTORY STOPPING CONDITION FOR ARC I2, SDINP
     1 4H AT A10,2H = E17.8 )                                            SDINP
  990 FORMAT(//,3X,19HCONSTRAINT FOR ARC I2,A10,2H =E17 8, 5H +OR-E17 8) SDINP
 1000 FORMAT(//,3X,33HWEIGHT DISCONTINUITY BEFORE  ARC I2, 2H =E17 8,  8 SDINP
     1H POUNDS.)                                                         SDINP
 1010 FORMAT(//,3X,26HBRANCH POINT AT END OF ARCI3,24H. FIRST BRANCH COM SDINP
     1PRISESI3,14H ARCS, AND HASI2,22H TERMINAL CONSTRAINTS.)            SDINP
 1020 FORMAT(//,3X,    14HTRAJECTORY HASI3,6H ARCS./,3XA10,10H IS TO BE , SDINP
     1A10,  36H  OPTIONAL GUESSED VALUE OF PAYOFF =E17 8   /)            SDINP
 1030 FORMAT(20(1H*),36HFIRST NOMINAL TRAJECTORY PHASE INPUT,20(1H*)//)  SDINP
C  PAYOFF MULTIPLY DEFINED                                               SDINP
 1040 I=1                                                                SDINP
 1050 JOP=MPSI(ILV)                                                      SDINP
      TALOP= SPSI(ILV,TARG(IK+2))                                        SDINP
      WRITE(6,1120)  INPERR,IOP,VALOP,JOP,TALOP                          SDINP
      IF(I-1)210,210,470                                                 SDINP
C   WEIGHT  DISCONTINUITY ERROR                                          SDINP
 1060 WRITE(6,1130)  INPERR,I                                            SDINP
      GO TO 380                                                          SDINP
C     INITIAL COND CODE INCONSISTENT WITH PRIOR STOPPING VARIABLE        SDINP
 1070 WRITE(6,1140)  INPERR,I,IVD,ICD                                    SDINP
      GO TO 380                                                          SDINP
C     BRANCH  PROBLEM AS  903                                            SDINP
 1080 WRITE(6,1150) INPERR,NSB,JST(NSB),I,IVD,ICD                        SDINP
      GO TO 380                                                          SDINP
C     IC  CODE  IS NOT INTERPRETABLE                                     SDINP
 1090 WRITE(6,1160) INPERR,I,IVD,ICD                                     SDINP
      GO TO 380                                                          SDINP
C     STOPP  COND  UNDEF                                                 SDINP
 1100 WRITE(6,1170) INPERR,I                                             SDINP
      GO  TO 500                                                         SDINP
C     STOP  COND MULT  DEFINED                                           SDINP
 1110 WRITE(6,1180) INPERR,I                                             SDINP
      IF(I-1) 160,160,470                                                SDINP
 1120 FORMAT( 2A10/,42H   PAYOFF MULTIPLY DEFINED,ORIGINALLY IOP=I3,     SDINP
     1 7H VALOP=E17 8,15H  NEW DATA IOP=I3, 7H VALOP=E17.8)              SDINP
 1130 FORMAT(2A10/,39H  WEIGHT DISCONTINUITY INDICATED IN ARCI3,34H AFTE SDINP
     1R IMPROPER STOPPING CONDITION)                                     SDINP
 1140 FORMAT(2A10/,29H INITIAL CONDTION CODE IN ARCI3,60H INCONSISTENT W SDINP
```

```
     1IIH PRIOR ARC STOPPING CONDITION VARIABLE NO=I3 , 6H CODE=I3)        SDINP
 1150 FORMAT(2A10/,33H BRANCH STOPPING CONDITION IN ARCI3, 6H ,JST=I3,     SDINP
     153H IS INCONSISTENT WITH BRANCH LEG INITIAL CONDITION IN/, 7H     AR SDINP
     2C=I3,15H. VARIABLE NO =I3, 6H,  CODE=I3)                             SDINP
 1160 FORMAT(2A10/,42H  INITIAL CONDITION CODE FOR ARCI3,18H, VARIABLE     SDINP
     1NUMBERI4,2H =I10,22H, IS NOT INTERPRETABLE)                          SDINP
 1170 FORMAT(2A10/,27H STOPPING CONDITION FOR ARCI3,13H IS UNDEFINED)      SDINP
 1180 FORMAT(2A10/,28H  STOPPING CONDITION FOR ARCI3,20H IS MULTIPLY DEF   SDINP
     1INED)                                                                SDINP
      END                                                                  SDINP
```

20 OCT 72 6.01-46

SUBRØUTINE
SØMG

FUNCTION SOMG

Entry SPSI

Purpose

Convert input external units to internal units. SOMG or SPSI is called from SDINP to convert the units of input boundary condition to internal units; i. e., degrees to radians and pounds to slugs-mass.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ER	E_R	I	Earth radius. (FT)	/GLOBAL/(	2)	COORDS	I	ER
						CRASH	I	REM
						EQUA3	I	ER
						GEINP	I	ER
						PADS1	I	ER
						PDBC	I	ER
						SOMG	I	ER
						TRTOSZ	I	ER
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FH3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						PDBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
RDI		I	Angle to radian conversion, .01745329252	/DATA /(	3)	BLICO	I	RDI
						DER3A	I	RDI
						FNTG	I	RDI
						GUI3A	I	RDI
						MODELA	I	RDI
						MODELB	I	RDI
						PADS1	D	RDI
						PROPB	I	RDI
						PROPIN	I	RDI
						REU3	I	RDI
						SDINP	I	RDI
						SOMG	I	RDI

```
      FUNCTION SOMG (MN, REZ)                                           SOMG
      COMMON/GLOBAL/                                                    GLOBAL
     *GR       ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM         GLOBAL
     *,JJOP(10) ,IFATAL   ,NARC      ,NBRAN     ,NFARC     ,ID(4)       GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG   ,MAXTAB                                GLOBAL
     *,GM      ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEOFL(20)       GLOBAL
     *,ITPSO ,KSOL     ,KGLOBL(8)                                       RETAP
      COMMON/DATA/                                                      DATA
     *PI       ,RAD       ,RDI       ,SC        ,UMF       ,TMPF      , DATA
     *FTNM     ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4        DATA
      ENTRY SPSI                                                        SOMG
      GO TO (10,10,20,10,30,20,20,20,10,10,10,10,20,20,20,10,10,20,20,20 SOMG
     *,10,10,10,20,10,10,20,10,10,10,10,10,10,10,10,10,10,10,10,10),MN  OS
   10 SOMG = REZ                                                        SOMG
      RETURN                                                            SOMG
   20 SOMG= REZ*RDI                                                     SOMG
      RETURN                                                            SOMG
   30 SOMG =REZ/GR                                                      SOMG
      RETURN                                                            SOMG
   40 SOMG = REZ/ER                                                     SOMG
      RETURN                                                            SOMG
      END                                                               SOMG
```

10 20 30

SUBRØUTINE SPLYNE

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
IFOB		I	Forward or adjoint integration flag = 1 means forward = 2 means adjoint	/XCODES/(	178)	ACCEL BEROCO BLYNE EQUA3 IMPUL SPLYNE TOPM	I I I I I I O	IFOB IFOB IFOB IFOB IFOB IFOB IFOB
LOCF		I	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the last value of the independent variable of the corresponding table.	/TABLE /(	1401)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP	M M I I I I O	LOCF Z LOCF Z LOCF Z Z
LOCI		I	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that points to the initial value of the independent variable of the corresponding table A zero entry indicates table not input.	/TABLE /(	1)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP THRUP	M M I I I I I O	LOCI X LOCI X LOCI X LOCI X
LOCL		M	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that indicates the last interval in which interpolation of the corresponding table occured.	/TABLE /(	701)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP	O M M I M I O	LOCL Y LOCL Y LOCL Y Y
NT		I	Largest univariant table number in this case.	/GLOBAL/(	66)	SPLICO SPLIZ SPLYNE	M I I	NT NT NT
X		I	A 50 word array that corresponds to tables 1 thru 50. Each entry is an integer that points to the initial value of the independent variable of the corresponding table A zero entry indicates table not input.	/TABLE /(	1)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP THRUP	M M I I I I I O	LOCI X LOCI X LOCI X LOCI X
Y		I	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that indicates the last interval in which interpolation of the corresponding table occured.	/TABLE /(	701)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP	O M M I M I O	LOCL Y LOCL Y LOCL Y Y
Z		I	A 50 word array that corresponds to tables 1 thru 50 Each entry is an integer that points to the last value of the independent variable of the corresponding table.	/TABLE /(	1401)	SPLICO SPLICO SPLIZ SPLIZ SPLYNE SPLYNE THRUP	M M I I I I O	LOCF Z LOCF Z LOCF Z Z

```
      SUBROUTINE SPLYNE(IT, T, F, DFDX)                                 SPLYNE
      COMMON /TABLE/ TABLE(2100) /GLOBAL/ G(66)                         SPLYNE
      DIMENSION X(1), Y(1), Z(1), LOCI(1), LOCL(1), LOCF(1)             SPLYNE
      EQUIVALENCE (X, TABLE), (Y, TABLE(701)), (Z, TABLE(1401)),        SPLYNE
     1(LOCI, X), (LOCL, Y), (LOCF, Z), (NT, G(66))                      SPLYNE
      COMMON /XCODES/                                                   XCODES
     *ITO     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST    (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
     *I20P      ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       , XCODES
     *IOPEN     ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
     *ITCT      ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
     *KOP       ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
     *NCN       ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         , XCODES
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN     ,NEQF  ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3        XCODES
      DATA SIXTH/17155252525252525253B/                                 SPLYNE
      DATA SPLINT /6HSPLINE/                                            SPLYNE
   10 FORMAT(1H0, 22HTHE TABLE NO. IN LOC  O6 17H IS OUT OF RANGE.)     SPLYNE
   20 FORMAT(1H0, 22HTHE TABLE NO  IN LOC. O6 20H HAS NOT BEEN INPUT.)  SPLYNE
      IF(IT .LE  NT) GO TO 30                                           SPLYNE
      ITLOC = XLOCF(IT)                                                 SPLYNE
      WRITE(6,10) ITLOC                                                 SPLYNE
      CALL STPIT (6)                                                    SPLYNE
   30 IF(IT  LE  0) GO TO 40                                            SPLYNE
      II = LOCI(IT)                                                     SPLYNE
      IF(II  GT  0) GO TO 50                                            SPLYNE
      ITLOC = XLOCF(IT)                                                 SPLYNE
      WRITE(6,20) ITLOC                                                 SPLYNE
      CALL STPIT (6)                                                    SPLYNE
   40 F = 0.                                                            SPLYNE
      DFDX = 0.                                                         SPLYNE
      RETURN                                                            SPLYNE
   50 IF = LOCF(IT)                                                     SPLYNE
      IF(IF - II - 1) 60,70,80                                          SPLYNE
   60 F = Y(II)                                                         SPLYNE
      DFDX = 0                                                          SPLYNE
      RETURN                                                            SPLYNE
   70 Y1 = Y(II)                                                        SPLYNE
      X1 = X(II)                                                        SPLYNE
      DFDX=(Y(IF) - Y1)/(X(IF) - X1)                                    SPLYNE
      F = Y1 + (T - X1)*DFDX                                            SPLYNE
      RETURN                                                            SPLYNE
   80 IF(T  GT. X(II)) GO TO 90                                         SPLYNE
      Y1 = Y(II)                                                        SPLYNE
      X1 = X(II)                                                        SPLYNE
      IIP1 =  II + 1                                                    SPLYNE
      DEL = X(IIP1) - X1                                                SPLYNE
      DFDX=(Y(IIP1) - Y1)/DEL - SIXTH*DEL*Z(IIP1)                       SPLYNE
      F = Y1 + (T - X1)*DFDX                                            SPLYNE
      RETURN                                                            SPLYNE
   90 IF(T  LT. X(IF)) GO TO 100                                        SPLYNE
      IFM1 = IF - 1                                                     SPLYNE
      Y1 = Y(IF)                                                        SPLYNE
      X1 = X(IF)                                                        SPLYNE
      DEL = X1 - X(IFM1)                                                SPLYNE
      DFDX=(Y1 - Y(IFM1))/DEL+ SIXTH*Z(IFM1)*DEL                        SPLYNE
      F = Y1 + (T - X1)*DFDX                                            SPLYNE
      RETURN                                                            SPLYNE
  100 IL = LOCL(IT)                                                     SPLYNE
      IF(T - X(IL)) 110,150,120                                         SPLYNE
  110 IL = IL - 1                                                       SPLYNE
      IF(T - X(IL)) 110,140,140                                         SPLYNE
  120 IS = IL + 1                                                       SPLYNE
      DO 130 I = IS, IF                                                 SPLYNE
      IF(T - X(I ) .LT. 0 ) GO TO 140                                   SPLYNE
  130 IL = I                                                            SPLYNE
  140 LOCL(IT) = IL                                                     SPLYNE
```

20 OCT 72 G 01-46

```
      150 ILP1 = IL + 1                                                 SPLYNE
          DX1 = T - X(IL)                                               SPLYNE
          DX2 = X(ILP1) - T                                             SPLYNE
          Y1 = Y(IL)                                                    SPLYNE
          Y2 = Y(ILP1)                                                  SPLYNE
          Z1 = Z(IL)                                                    SPLYNE
          Z2 = Z(ILP1)                                                  SPLYNE
          DEL = DX1 + DX2                                               SPLYNE
          TEMP1= Z2*DX1 + Z1*DX2                                        SPLYNE
          TEMP2 = SIXTH*DEL                                             SPLYNE
          F = ((Y2*DX1 + Y1*DX2) + SIXTH*(Z2*DX1**3 + Z1*DX2**3))/DEL   SPLYNE
         1 - TEMP2*TEMP1                                                SPLYNE
           IF(IFDB NE.2) RETURN                                         SPLYNE
          DFDX = ((Y2 - Y1) + .5*(Z2*DX1**2 - Z1*DX2**2))/DEL -TEMP2*(Z2-Z1) SPLYNE
          RETURN                                                        AAA
          END                                                           AAA
```

20 OCT 72 G 01-46

674

SUBRØUTINE STAU

Subroutine STAU

Entry SINIT, ADJUMP

Purpose

Subroutine STAU has three entry points. The main entry, STAU, computes the arc time sensitivity. Entry SINIT stores initial state sensitivity in the parameter sensitivity matrix, and prints the entire parameter sensitivity matrix at the completion of the adjoint solution.

Entry ADJUMP computes the adjoint discontinuity at the critical staging time for rubber stage optimization.

Description

The arc time sensitivity equation is given in equation (12.1-2) of Volume I. The rubber stage adjoint discontinuity is given in Section 13 of Volume I.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DWEB		I	Sensitivity of booster stage weight to propellant weight (lb/lb)	/SIZING/(	274)	PAYLOD	I	DWEB
						SIZOUT	I	DWEB
						SIZ1	M	DWEB
						SIZ2	M	DWEB
						STAU	I	DWEB
						WTDRP	M	DWEB
GR	g_r	I	Gravitational acceleration at surface of the earth. (FT/SEC^2)	/GLOBAL/(	1)	ACCEL	I	GR
						BL5	I	GR
						EQUA3	I	GR
						FM3	I	GR
						GEINP	I	G
						GEINP	I	GR
						GEINP	O	IG
						OUT	I	GR
						PADS1	I	GR
						POBC	I	GR
						REU3	I	GR
						SDINP	I	GR
						SIZE	I	GR
						SIZ1	I	GR
						SIZ2	I	GR
						SIZ3	I	GR
						SIZ4	I	GR
						SOMG	I	GR
						STAU	I	GR
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
IPOINT		I	Code for each adjustable parameter in steepest descent.	/PARAM /(	1)	ADJUST	I	IPOINT
						PRMSET	I	IPOINT
						SDINP	O	IPOINT
						STAU	I	IPOINT
						TOPM	D	IPOINT
ITQ		I	Constraint option code (internal)	/XCODES/(	1)	ADICB3	I	ITQ
						ADIC3A	I	ITQ
						ADID3A	I	ITQ
						CON3	I	ITQ
						SDINP	M	ITQ
						STAU	I	ITQ
						TOPM	D	IITQ
NEQF		I	Number of equations to be integrated on forward trajectory	/XCODES/(	185)	REU3	I	NEQF
						RKTA3A	I	NN
						SDINP	O	NEQF
						STAU	I	NEQF
						TOPM	O	NEQF
						TRAN3	O	NEQF
NPARA		I	Number of adjustable parameters in trajectory problem.	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	D	NPARA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
SPARA	S^{Ψ_i}	M	Matrix of adjustable parameter sensitivities (including all parameters)	/PARAM /(	15)	ADJUST	I	SPARA
						PAYO2	I	SPARA
						STAU	M	SPARA
						TOPM	D	SPARA
SQ		I	A synthesis data array (37,5) that contains the flyback data and some injection quantities	/SIZING/(	74)	ENVPRM	M	SQ
						FLYBKP	M	SQ
						ISPRAT	I	SQ
						PDBC	I	SQ
						PRITVA	I	SQ
						RANGE	M	SQ
						REU3	D	SQ
						SIZE	0	SQ
						SIZEMR	M	SQ
						SIZIN	M	SQ
						STAU	I	SQ
						SUMOUT	M	SQ
						TAMPAR	0	SQ
						TAMPER	M	SQ
						THRUST	M	SQ
						TRTOSZ	M	SQ
						VEHDF	M	SQ
						WTVOL	M	SQ
XL	$\lambda^{\Psi_j \Omega_J}$	M	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	0	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
YDS	$\dot{y}\vert_{\tau-}$	I	Array of state derivatives at arc end points	/STATE3/(	507)	ADICB3	I	YDS
						ADID3A	I	YDS
						REU3	0	YDS
						STAU	I	YDS

```
      SUBROUTINE  STAU( MN,NM)                                          STAU
C       THIS  ROUTINE  COMPUTES   STAGING  TIME  SENSITIVITIES          STAU
      COMMON/STATE3/                                                    STATE30
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10)  , STATE30
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE30
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   , STATE30
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                 STATE30
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE30
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE30
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE30
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE30
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE30
     *PDO        ,UDO       ,HTDV      ,HTDR                            STATE30
      REAL MDM , MDV, MDR                                               STATE30
      COMMON/STATE3/                                                    STATE30
     *SIN2RO     ,COS2RO    ,COS2GM                                     STATE30
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE    , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON/PARAM/                                                     PARAM
     *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),           PARAM
     *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                       PARAM
     *,DELP(9)                                                          PARAM
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARO(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,OPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                     GENF
     *LIFTR      ,LIFTA     ,          DRAGV      ,DRAGR     ,DRAGA    , GENF
     *           LIFTM      ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                      ULFT       ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                   GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED     ,IRATED    ,                                           GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,               MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      REAL MUB, MUO, ISPB, ISPO, IDVEL,NNB,NO                           SIZING
      COMMON /SIZING/                                                   SIZING
C           PHASE II SIZING PARAMERERS                                  SIZING
     *TZ,      VV(3),      QP(14),    EROR,      PZ(5),     VQ,      SW(20), SIZING
     *SV(28),   SQ(37,5),  SE(11),    TLAT,      TLNG,                  SIZING
C          PHASE I SIZING PARAMERERS                                    SIZING
     *WBO,      WLOO,      OWEB,      OWEO,      TOLWT,     WPB,     TWRAT2, SIZING
     *BK1,      BK2,       BK3,       BK4,       ISIZE,     TRAFLG,  TWRATO, SIZING
     *OK1,      OK2,       OK3,       OK4,       PRFLG,     IPASS,   IPSMAX, SIZING
     *AEXIT,    TVACO,     NO,        WFO,       IDVEL,     ISPO,    ISPB,  SIZING
     *XPL,      TVACB,     NNB,       WEO,       WEB,       WO,      WLO,   SIZING
```

20 OCT 72 G 01-46

```
      *DVJ,      DVB,      MU5,      MU0,      VSTG,     WFD          SIZING
      *,JTYP,    BECO ,    BSTG ,    ORBI,     ITNBW , ITNOW  ,       SIZING
      * SVOPSQ  , SVOCON  ,THUNT   ,IOPSTG  ,ISZO(19)                 UM
      COMMON/GLOBAL/                                                  GLOBAL
      *GR        ,ER        ,OMGZ     ,XLAMRF    ,YMURF     ,LUM      GLOBAL
      *,JJOP(10) ,IFATAL    ,NARC      NBRAN     ,NFARC     ,ID(4)    GLOBAL
      *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                            GLOBAL
      *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEOFL(20) GLOBAL
      *,ITPSD ,KSOL     ,KGLOBL(8)                                    RETAP
C     I                                                               STAU
C        TEST  FOR  PRESENCE  OF  OPTIMAL  STAGE  TIME  IN  POINTER   STAU
C        ARRAY                                                        STAU
      IIS =IARC-1                                                     STAU
      DO 10 I=1,NPARA                                                 STAU
      JJ = I                                                          STAU
      IF(IPOINT(I).EQ.IIS )   GO TO 20                                STAU
   10 CONTINUE                                                        STAU
      RETURN                                                          STAU
C      II    STAGING  TIME  SENSITIVITY                               STAU
   20 DO  40  I=MN,NN                                                 STAU
      SS =0.                                                          STAU
      DO  30  J=1,NEQF                                                STAU
      SS = SS   + XL(J,I)* YDS(IIS,J)                                 STAU
   30 CONTINUE                                                        STAU
      SPARA(I,JJ)= SS                                                 STAU
   40 CONTINUE                                                        STAU
      RETURN                                                          STAU
C           OPTIMAL  INITIAL  CONDITIONS                              STAU
      ENTRY  SINIT                                                    STAU
C  III     TEST  FOR  PRESENCE  OF  OPTIMAL  IC  AND COMPUTE SENSITIVIT STAU
      DO 60 I=1,NPARA                                                 STAU
      JJ = I                                                          STAU
      IF(IPOINT(I).LE.21) GO TO 60                                    STAU
C  IIIA   COMPUTE  SENSITIVITY                                        STAU
      JI= IPOINT(JJ) - 21                                             STAU
      IF(JI.GT.7) GO TO 60                                            STAU
      DO 50  J=MN,NN                                                  STAU
   50 SPARA(J,JJ) = XL(JI,J)                                          STAU
   60 CONTINUE                                                        STAU
      WRITE(6,70)((SPARA(I,J),I=MN,NN),J=1,NPARA)                     STAU
   70 FORMAT(28H0   PARAMETER SENSITIVITIES   /(6E20.8))              STAU
      RETURN                                                          STAU
C        IV   ADJOINT  DISCONT  FOR  VARIABLE  WT- DROPPED            PHISZ
      ENTRY  ADJUMP                                                   PHISZ
      DPLDWB = SQ(4,1)*GR                                             JULY28
      DO 90 I=MN,NN                                                   OS
      IF(ITQ(I).NE.36) GO TO 80                                       OS
      XL(4,I) = XL(4,I)*(1 -DWEB) + DPLDWB                            JULY28
      GO TO 90                                                        OS
   80 CONTINUE                                                        OS
      XL(4,I) = XL(4,I)*(1 -DWEB)                                     JULY28
   90 CONTINUE                                                        OS
      RETURN                                                          PHISZ
      END                                                             STAU
```

20 OCT 72 6.01-46

6-36

SUBRØUTINE STPIT

Subroutine STPIT

Purpose

Subroutine STPIT is called for fatal errors in steepest descent execution.

Description

This error routine prints a number message (number comes through argument list) and returns to PADS through a call to entry point SDERR therein.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
IFATAL		0	Fatal error flag.	/GLOBAL/(	17)	BLICO	M	IFATAL
						GEINP	O	IFATAL
						PADSI	I	IFATAL
						SDINP	M	IFATAL
						SPLICO	M	IFATAL
						STPIT	O	IFATAL
						TOPM	M	IFATAL
.UN06.		0	File of all output data	/.UN06./($	)	BLICO	O	.UN06.
						BNDRYC	O	.UN06.
						CRASH	O	.UN06.
						FRENCH	O	UN06.
						FXDAT	O	.UN06.
						GEINP	O	UN06
						HUNT	O	.UN06.
						INEDIT	O	.UN06.
						ITER8	O	UN06
						MODELA	O	UN06.
						MOMJ	O	.UN06.
						MPSI	O	.UN06.
						OUT	O	.UN06.
						PAY02	O	.UN06.
						PRINT	O	.UN06.
						PRINTV	O	.UN06.
						PRINTW	O	.UN06.
						PRITEQ	O	UN06.
						PRITVA	O	.UN06
						PROPIN	O	.UN06.
						PROTHR	O	.UN06.
						PRWTSM	O	.UN06.
						RANGE	O	.UN06.
						S	O	.UN06.
						SDINP	O	.UN06.
						SIZE	O	.UN06.
						SIZIN	O	.UN06.
						SIZOUT	O	.UN06.
						SOLVE	O	.UN06.
						SPLICO	O	.UN06.
						SPLIZ	O	.UN06.
						SRLYNE	O	.UN06.
						SSSP	O	.UN06.
						STAU	O	.UN06.
						STPIT	O	.UN06.
						SUMOUT	O	.UN06.
						TABIN	O	.UN06.
						TEST	O	UN06.
						VEHDF	O	.UN06.
						WTSCH	O	.UN06.
						WTVOL	O	.UN06.

```
      SUBROUTINE STPIT(I)                                               STPIT
C                                                                       COMN
C       ERROR NUMBER PRINT  AND  RETURN  TO  MAIN  PROGRAM              COMN
C                                                                       COMN
      COMMON/GLOBAL/                                                    GLOBAL
     *GR        ,ER         ,OMGZ      ,XLAMRF     ,YMURF     ,LUM      GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC       ,NBRAN     ,NFARC     ,ID(4)    GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                               GLOBAL
     *,GM          ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEOFL(20)  GLOBAL
     *,ITPSO ,KSOL     ,KGLOBL(8)                                       RETAP
      LOGICAL IFATAL                                                    UH
      WRITE(6,10) I                                                     UH
   10 FORMAT(40(1H*),I3)                                                STPIT
      IFATAL= TRUE                                                      UH
      CALL SDERR                                                        UH
      END                                                               STPIT
```

20 OCT 72 G 01-46

SUBRØUTINE
STP3

Subroutinc STP3

Entry Points. PHSTOP, STSTOP, SETFPO

Purpose

This routıne detects whether arc or phase cut-off will be reached in next integration interval on forward trajectory.

Description

The logic and equations for this routine are described in Section 15.2 of Volume I. This routine as called from FNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DT		I	Integration interval (SEC)	/GENF /(	300)	BNTG	M	DT
						FNTG	M	DT
						REU3	I	DT
						RKTA3A	I	P
						RKTB3A	I	P
						STP3	I	DT
						YREF3	O	DT
DVAR	$\dot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
FP		I	Current value of cut-off function - non-linear only	/GENF /(	521)	DTF3	I	FP
						STP3	I	FP
						TOL3	I	FP
						YREF3	I	FP
FPD		I	Rate of change of non-linear cut-off function	/GENF /(	523)	CON3	I	FPD
						DTF3	I	FPD
						STP3	I	FPD
						YREF3	I	FPD
FPOLD		O	Value of non-linear cut-off function at prior compute interval	/GENF /(	522)	DTF3	I	FPOLD
						STP3	O	FPOLD
						TOL3	O	FPOLD
						YREF3	I	FPOLD
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
INEQFL		M	A 20 word array that contains the code number of the state variables inequality constraint that applies on each subarc A zero entry indicates that no SVIC applies.	/GLOBAL/(	73)	PROPB	I	INEQFL
						PROPIN	M	INEQFL
						STP3	M	INEQFL
ISPH		I	Sign of phase cut-off	/XCODES/(	144)	FNTG	O	ISPH
						STP3	I	ISPH
ISST		I	Sign of arc cut-off	/XCODES/(	145)	FNTG	O	ISST
						STP3	I	ISST
ISTOP		I	Arc cut-off flag	/XCODES/(	175)	FNTG	M	ISTOP
						STP3	I	ISTOP
ISTPP		I	Phase cut-off flag	/XCODES/(	176)	FNTG	M	ISTPP
						STP3	I	ISTPP
IVAR		O	Cut-off variable option indicator	/XCODES/(	150)	FNTG	M	IVAR
						STP3	O	IVAR
						TOL3	I	IVAR

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
JPS		I	Absolute value of phase cut-off option code		/XCODES/(	152)	ADID3A	I	JPS
							BNTG	M	JPS
							FNTG	M	JPS
							STP3	I	JPS
							TOL3	I	JPS
JS		I	Absolute value of arc cut-off option code		/XCODES/(	153)	ADICB3	M	JS
							ADIC3A	I	JS
							ADID3A	I	JS
							BNTG	M	JS
							FNTG	M	JS
							PROPB	I	JS
							PROPIN	I	JS
							STP3	I	JS
							TOL3	I	JS
OMG	Ω_J	I	Array of arc cut off values [sd]		/GENF /(	1)	ADJUST	M	OMG
							FNTG	I	OMG
							PRMSET	M	OMG
							PROPB	I	OMG
							SDINP	M	OMG
							STP3	I	OMG
							TOPM	D	IOMG
OMP		I	Phase cut-off value		/GENF /(	494)	ADJUST	O	OMP
							FNTG	M	OMP
							STP3	I	OMP
VAR	v	I	Relative velocity	(FT/SEC)	/STATE3/(	1)	ACCEL	I	V
							ADICB3	O	VAR
							ADJUST	M	VAR
							AGETB3	O	VAR
							AST3	I	VAR
							BL4	I	V
							BL7	I	V
							BL8	I	V
							CON3	I	VAR
							DER3A	I	V
							DTF3	I	V
							ENVPRM	I	VAR
							EQUA3	I	V
							MODELA	I	V
							MODELA	I	VAR
							MODELB	I	V
							MTX3A	I	VAR
							OUT	I	V
							OUT	I	VAR
							PDBC	I	V
							PDY3A	I	V
							REU3	M	VAR
							RKTA3A	M	V
							STP3	I	VAR
							TOPM	D	KWDW
							VREF3	M	V

```
      SUBROUTINE STP3(IVR  )                                         STP3
C                                                                    CORN
C        DETECTS  WHETHER  CUT-OFF  WILL  BE REACHED IN  NEXT        CORN
C        INTEGRATION  INTERVAL  (FORWARD TRAJECTORY)                 CORN
C          FLAG IVR=1 =CUT-OFF WILL OCCUR , IVR=0= WILL NOT OCCUR    CORN
C                                                                    CORN
      COMMON/STATE3/                                                 STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10) , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,DCORHO    ,DCORO2  , STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                 STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV     , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG     , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR     , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP     , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO     , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                            STATE3D
      REAL MDM , MDV, MDR                                            STATE3D
      COMMON/STATE3/                                                 STATE3D
     *SIN2RO     ,COS2RD    ,COS2GM                                  STATE3D
      COMMON/GENF/                                                   GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TDL(9)    ,SVAR(10)  ,WDC(20) , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP     , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS      , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS      , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ   , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TDP       ,TDS       ,TR(9)   , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W       , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP     , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20) , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV   , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                     GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA   , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF    , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA   , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE   , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID     , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                   GENF
      COMMON / GENF  /                                               GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP     , GENF
     *FRATED    ,IRATED     ,                                           GENF
     *P1         ,P2        ,P3       ,XKI        ,XK2       ,XK3     , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D    , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V    , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P    , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O    , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M    , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8) GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,               MACH, MACHR,  GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                               FRAT
      DIMENSION TPH1(10),TST1(10)                                    GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                             GENF
      COMMON /XCODES/                                                XCODES
     *ITQ    (9),ICOR  (20),ITI        ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB   , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND     , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART  , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS      , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE   , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N       , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB   , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L       , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB    , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3       XCODES
      COMMON/GLOBAL/                                                 GLOBAL
     *GR         ,ER        ,OMGZ      ,XLAMRF     ,YMURF     ,LUM      GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC        ,NBRAN     ,NFARC     ,IDI4)   GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                               GLOBAL
     *,GM         ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)    GLOBAL
     *,ITPSO ,KSOL      ,KGLOBL(8)                                      RETAP
      DIMENSION SX(8)                                                STP3
      DIMENSION RATE(3)                                              APR27
      DATA  RATE/1.5,0.,0./                                          APR27
      ENTRY PHSTOP                                                   STP3
      IF(JPS GE.10) GO TO 30                                         STP3   30
```

```
C        I     PHASE   CUT-OFF                                              COMN
C        I-A        ON   STATE                                              COMN
      IF( ISPH*( VAR(ISTPP) +DT*DVAR(ISTPP)-OMP   )) 20,10,10               STP3
   10 IVR= 1                                                                STP3
      IVAR = ISTPP                                                          STP3
      RETURN                                                                STP3
   20 IVR =0                                                                STP3
      RETURN                                                                STP3
C        I-B   PHASE  CUT-OFF  ON  FUNCTION  OF  STATE                      COMN
   30 MD=0                                                                  STP3
      CALL PDBC(JPS,FP,SX,SFD,0,MD)                                         STP3
       MD=0                                                                 STP3
      CALL PDBC(JPS,DVAR,SX,FPD,3,MD)                                       STP3
      IF(ISPH*(FP+DT*FPD-OMP)) 40,10,10                                     STP3
   40 FPLD=FP                                                               APR27
      GO TO 20                                                              STP3
C        II    ARC  CUT-OFF  TEST                                           COMN
      ENTRY STSTOP                                                          STP3
      IF(JS .GE.10) GO TO 60                                                STP3
C        II-A    ON  STATE                                                  COMN
      IF( ISST*( VAR(ISTOP) + DT*DVAR(ISTOP)-OMG(IARC)  ))20,50,50          STP3
   50 IVR =1                                                                STP3
      IVAR = ISTOP                                                          STP3
      RETURN                                                                STP3
C        II-B     ON  FUNCTION OF STATE                                     COMN
   60 MD=0                                                                  STP3
      CALL PDBC(JS,FP,SX,SFD,0,MD)                                          STP3
      MD=0                                                                  STP3
      CALL PDBC(JS,DVAR,SX,FPD,3,MD)                                        STP3
C        II-C    IF  NOT AN INEQUALITY  CUT-OFF  GO TO II-E                 COMN
      IF(INEQFL(IARC) EQ 0) GO TO  70                                       STP3
C        II-B   ADJUST  INEQUAITY FUNCTION RATE  BY TOLERANCE               COMN
C        AND   TEST  FOR  ZERO   IF LESS  THAN ZERO  GO TO II-F             COMN
      JJ=JS-32                                                              APR27
      IF(FPD-RATE(JJ)) 90,90,70                                             APR27
   70 CONTINUE                                                              STP3
C        II-E   TEST FOR  CUT-OFF  YES, SET FLAG, NO SAVE FUNCTION          COMN
      IF(ISST*(FP+DT*FPD-OMG(IARC))) 80,50,50                               STP3
   80 FPOLD= FP                                                             STP3
      GO TO 20                                                              STP3
C        II-F   INEQUALITY   CUT-OFF  NOT  REACHED,                         COMN
C               DERIVATIVE  ZERO, SET  FLAGS                                COMN
   90 IVR=1                                                                 STP3
      IVAR =-JS                                                             STP3
      INEQFL(IARC)= 0                                                       STP3
      RETURN                                                                STP3
      ENTRY SETFPD                                                          APR27
       FPOLD=FPLD                                                           APR27
      RETURN                                                                APR27
      END                                                                   STP3
```

10 20
40 10
20
60
50 20
70
70 90
80 50
20

694

SUBRØUTINE STVRL3

Subroutine STVRL3

Entry. STVRL

Purpose

STVRL resets adjoint integration variables after discontinuities.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NB		I	Extent of integration set during adjoints on branch problem	/XCODES/(	179)	ADEQ3A ADICB3 ADIC3A BNTG RKTB3A STVRL3	I M M O I I	NB NB NB NB NB NB
VARL		O	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A ADICB3 ADIC3A PROPIN RKTA3A RKTB3A STVRL3 TRAN3	I O O I M M O M	VARL VARL VARL ZZ F Y VARL VARL
XL	$\lambda^{\Psi_i \Omega_j}$	I	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A ADICB3 ADIC3A ADID3A AST3 BGET3 BSTO3 MTX3A OUT STAU STVRL3 TRAN3	M M M M M O I I I M I M	XL XL XL XL XL XL XL XL XL XL XL XL

30 OCT 72 G 01-46

STVRL3

```
      SUBROUTINE STVRL3(LL,MM)                                          STVRL3
C                                                                       COMN
C       RESETS  ADJOINT  INTEGRATION  VARIABLES AFTER                   COMN
C       DISCONTINUITIES                                                 COMN
C                                                                       COMN
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)      ,SVY(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM     ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO     ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2    ,                                STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV        ,ODV      , STATE3D
     *UDV        ,VDG       ,GDG       ,ROG       ,PDG        ,ODG      , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR        ,ODR      , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM        ,VDP      , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO        ,GDO      , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                            STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                     STATE3D
      COMMON /XCODES/                                                   XCODES
     *ITQ    (9),ICOR  (20),ITI        ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB       ,NICNB    , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB        ,IND      , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC       ,ISTART   , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS        ,JS       , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD        ,NCASE    , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH        ,N        , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN       ,ISTNB    , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP      ,L        , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP       ,NPHB     , XCODES
     *NCTIN      ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      ENTRY STVARL                                                      STVRL3
      JJ = NB                                                           STVRL3
      DO 20  IK=LL,MM                                                   STVRL3
      DO 10 IJ=1,6                                                      STVRL3
      VARL(JJ)= XL(IJ,IK)                                               STVRL3
   10 JJ=JJ+1                                                           STVRL3
   20 CONTINUE                                                          STVRL3
      RETURN                                                            STVRL3
      END                                                               STVRL3
```

20 OCT 72 G 01-46

SUBRØUTINE
SUMS

FUNCTION SUMS

Purpose

The purpose of SUMS is to compute the weighted sum of squares of constraint misses for use in determining constraint miss convergence.

Description

SUMS is called from TEST.

966

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BSTO3	I	NCNST
						CON3	I	NCNST
						PAYO2	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST
TOL		I	Tolerence on constraint misses [sd]	/GENF /(	70)	SDINP	M	TOL
						SUMS	I	TOL
						TEST	I	TOL

697

SUMS

```
      FUNCTION SUMS (DC )                                               SUMS
C                                                                       COMN
C        COMPUTES   SUM OF SQUARES OF CONSTRAINT  MISSES                COMN
C        FOR USE  IN  DETERMINING   CONTRAINT  MISS  CONVERGANCE        COMN
      DIMENSION DC(1)                                                   SUMS
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV                            GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED     ,IRATED    ,                                           GENF
     *P1         ,P2        ,P3      ,XK1         ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE    , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3        XCODES
      SUMS =0.                                                          SUMS
      DO 10  I=1,NCNST                                                  SUMS
   10 SUMS = SUMS + (DC(I)/TOL(I))**2                                   SUMS
      RETURN                                                            SUMS
      END                                                               SUMS
```

20 OCT 72 G.01-46

SUBRØUTINE
SYMVRT

Subroutine SYMVRT

Purpose

This is a utility symmetric matrix inversion routine.

Description

SYMVRT is used to invert the "A" matrix as in equation 12.1-25 of Volume 1. SYMVRT is called from several routines including MTX, PAYO2, and TRAN 3. The method is described in E. Rodewig's Matrix Calculus (North Holland Publishing Co., Amsterdam, 1956), pp. 110-114.

```
      SUBROUTINE SYMVRT(A,N,ISING)                                  SYMVRT
C                                                                   COMM
C        SYMMETRIC  MATRIX  INVERSION ROUTINE                       COMM
C        USES   CHOLESKY  ALGORITHM                                 COMM
      DIMENSION A( 9, 9),B(9 ),T( 9)                                SYMVRT
      IX=N                                                          SYMVRT
      ISING=0                                                       SYMVRT
      DO 90 ID=1,IX                                                 SYMVRT
      T(ID)=1.0                                                     SYMVRT
      IDM1=ID-1                                                     SYMVRT
      IDP1=ID+1                                                     SYMVRT
      IF(IDM1)10,30,10                                              SYMVRT
   10 DO 20 J=ID,IX                                                 SYMVRT
      DO 20 K=1,IDM1                                                SYMVRT
   20 A(ID,J)=A(ID,J)-A(K,ID)*A(K,J)*T(K)                           SYMVRT
   30 IF(A(ID,ID))50,40,60                                          SYMVFT
   40 ISING=ID                                                      SYMVRT
      GO TO 210                                                     SYMVRT
   50 T(ID)=-1.0                                                    SYMVRT
   60 A(ID,ID)= SQRT(ABS(A(ID,ID))) *T(ID)                          SYMVRT
      IF(IDP1-IX)70,70,100                                          SYMVRT
   70 DO 80 J=IDP1,IX                                               SYMVRT
   80 A(ID,J)=A(ID,J)/A(ID,ID)                                      SYMVRT
      A(ID,ID)=A(ID,ID)*T(ID)                                       SYMVRT
   90 CONTINUE                                                      SYMVRT
  100 A(IX,IX)=A(IX,IX)*T(IX)                                       SYMVRT
      IXM1=IX-1                                                     SYMVRT
      DO 120 I=1,IXM1                                               SYMVRT
      I1=I+1                                                        SYMVRT
      DO 110 J=I1,IX                                                SYMVRT
  110 A(I,J)=A(I,J)/A(I,I)                                          SYMVRT
  120 A(I,I)=T(I)/A(I,I)                                            SYMVRT
      A(IX,IX)=T(IX)/A(IX,IX)                                       SYMVRT
      DO 150 I=2,IX                                                 SYMVRT
      K=I+1                                                         SYMVRT
      IM1=I-1                                                       SYMVRT
      DO 150 L=1,IM1                                                SYMVRT
      IF(K-IX)130,130,150                                           SYMVRT
  130 DO 140 J=K,IX                                                 SYMVRT
  140 A(L,J)=A(L,J)-A(I,J)*A(L,I)                                   SYMVRT
  150 A(L,I)=-A(L,I)*A(I,I)                                         SYMVRT
      DO 180 I=1,IX                                                 SYMVRT
      B(I)=0.0                                                      SYMVRT
      DO 160 K=I,IX                                                 SYMVRT
  160 B(I)=B(I)+A(I,K)**2*T(K)                                      SYMVRT
      M=I+1                                                         SYMVRT
      IF(M-IX)170,170,190                                           SYMVRT
  170 DO 180 J=M,IX                                                 SYMVRT
      A(J,I)=0.0                                                    SYMVRT
      DO 180 K=J,IX                                                 SYMVRT
  180 A(J,I)=A(J,I)+A(I,K)*A(J,K)*T(K)                              SYMVRT
  190 DO 200 I=1,IX                                                 SYMVRT
  200 A(I,I)=B(I)                                                   SYMVRT
  210 RETURN                                                        SYMVRT
      END                                                           SYMVRT
```

20 OCT 72 G.01-46

701

SUBRØUTINE TBLK

Subroutine TBLK

Purpose

Subroutine TBLK is a single-argument table lookup using linear interpolation.

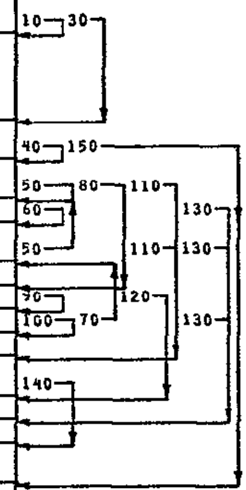
Description

This linear interpolation routine is employed for starting-nominal control tables. It saves the index where the last lookup occurred. TBLK is called from GUID3.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
AR		I	Array for storing starting control history tables	/STS	/(112)	SDINP SDINP TBLK	I I I	AR III AR
IAD		I	Starting address of each control history table	/STS	/(312)	SDINP TBLK	M I	IAD IAD
INP		I	Index of last argument of each control history table	/STS	/(332)	SDINP TBLK	M I	INP INP
ISV		M	Saved index of last control history table look-up	/STS	/(352)	SDINP TBLK	D M	ISV ISV

TBLK

```
      SUBROUTINE TBLK(ICV,ARG,ANS)                                      TBLK
      COMMON/STS/                                                       STS
     *DPAY      ,PMIN      ,WORK (20),NWDS      ,IPC    (7),NITER     , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)           STS
C SINGLE ARGUMENT TABLE  LOOKUP WITH SAVED INDEX FOR ASCENDING ARGUMENT TBLK
      IF(ICV) 30,10                                                     TBLK
   10 PRINT 20                                                          TBLK
   20 FORMAT(19H DATA CURVE MISSING)                                    TBLK
      II=LOCF(ICV)                                                      TBLK
      CALL IPR(7HLABELOC,K,II,1,1)                                      TBLK
      STOP                                                              TBLK
   30 CONTINUE                                                          TBLK
      IF(IAD(ICV).EQ.INP(ICV)) 150,40                                   TBLK
   40 K= ISV(ICV)                                                       TBLK
      IF(ARG -AR(K)) 50,110,80                                          TBLK
   50 IF((K-2).LT IAD(ICV))130,60                                       TBLK
   60 K=K-2                                                             TBLK
      IF( AR(K)  -ARG)130,110,50                                        TBLK
   70 K=K+2                                                             TBLK
   80 IF((K+2).GT.INP(ICV)) 120,90                                      TBLK
   90 IF(AR(K+2)- ARG)70,100,130                                        TBLK
  100 K=K+2                                                             TBLK
  110 ANS= AR(K+1)                                                      TBLK
      GO TO 140                                                         TBLK
  120 K=K-2                                                             TBLK
  130 ANS= AR(K+1) + (ARG -AR(K))*(AR(K+3)-AR(K+1))/(AR(K+2)- AR(K))    TBLK
  140 ISV(ICV)= K                                                       TBLK
      RETURN                                                            TBLK
  150 K=IAD(ICV)                                                        TBLK
      ANS=AR(K)                                                         TBLK
      RETURN                                                            TBLK
      END                                                               TBLK
```



20 OCT 72 G.01-46

SUBRØUTINE
TEST

TEST

Purpose

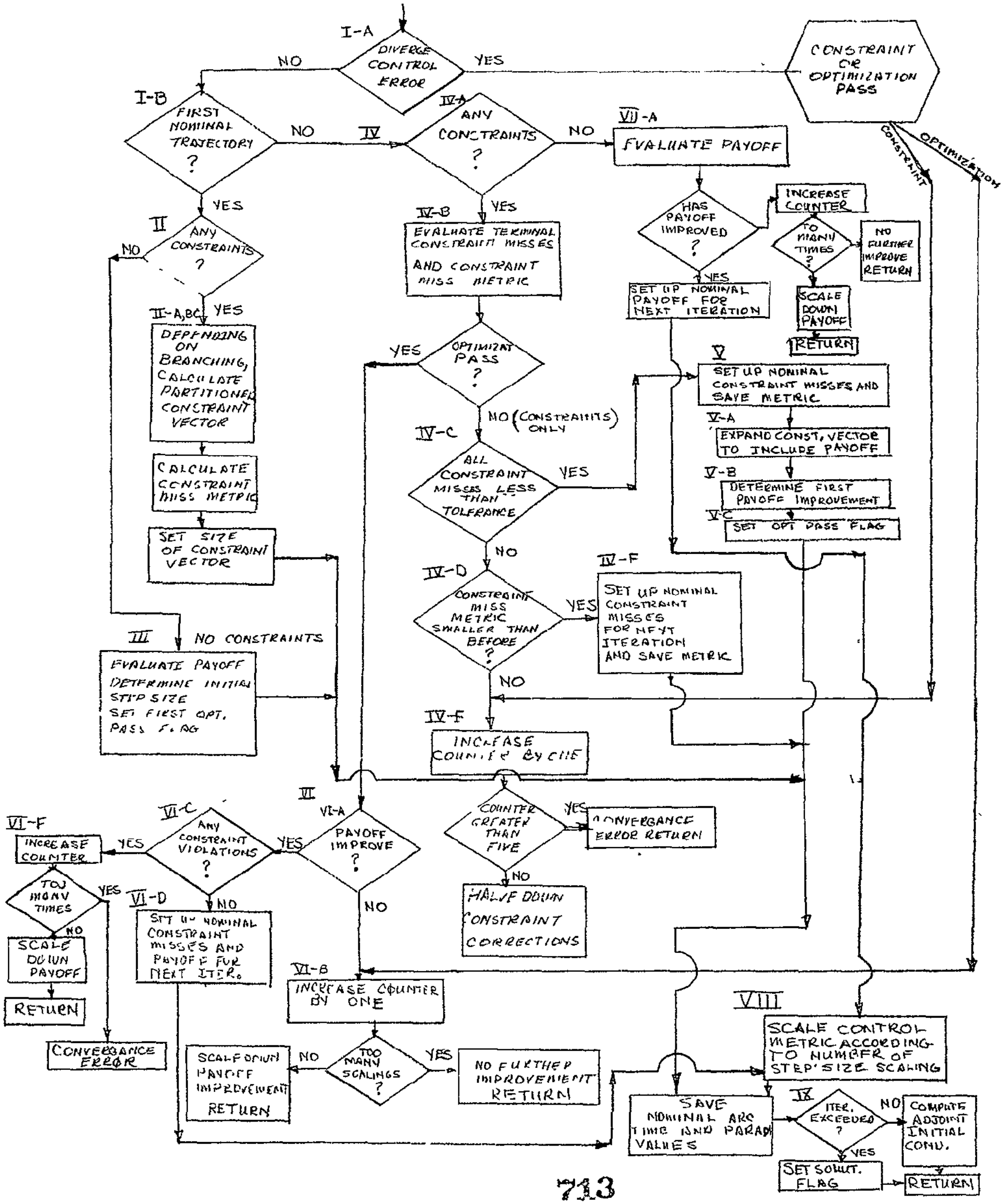
TEST evaluates constraint miss and performance improvement convergance.

Description

TEST is called by TOPM (steepest descent executive routine) at the completion of each trial trajectory to determine how well solution convergance is going. When difficulties arise, TEST detects them and prints out appropriate messages. Corrective action usually is a successive step size scaling process. When difficulties are not present, TEST calls ADIC to initialize the adjoints for the next adjoint solution-iteration sequence.

TEST

I TEST FLAGS



702

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
ACON		M	Vector of nominal constraint misses + PAYOFF IMPROVEMENT	/GENF /(	190)	CON3	M	ACON
						TEST	M	ACON
BCON		I	Vector of constraint misses on trial trajectory	/GENF /(	199)	CON3	O	BCON
						TEST	I	BCON
						TOPM	I	BCON
DCON	$d\Psi_i$	M	Asked for correction in constraint misses and payoff vector	/GENF /(	289)	CON3	O	DCON
						MTX3A	I	DCON
						PAYO2	M	DCON
						TEST	M	DCON
						TOPM	I	DCON
						TRTOSZ	I	DCON
DIP1		O	Phase initial times for nominal trajectory [sd]	/GENF /(	453)	GETIT	I	DIP1
						SDINP	M	DIP1
						TEST	O	DIP1
						TOPM	I	DIP1
DIS1		O	Arc initial times for nominal trajectory [sd]	/GENF /(	473)	GETIT	I	DIS1
						SDINP	M	DIS1
						TEST	O	DIS1
						TOPM	I	DIS1
						TRAN3	I	DIS1
DPAY	$d\Phi$	I	Initial payoff improvement	/STS /(	1)	PAYO2	I	DPAY
						SDINP	I	DPAY
						SDINP	O	IST
						SDINP	I	ST
						TEST	I	DPAY
						TOPM	D	IDPAY
INTB		I	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
						BNTG	I	INTB
						ENVPRM	I	INTB
						FNTG	I	INTB
						SDINP	M	INTB
						TEST	I	INTB
						TRAN3	I	INTB
						TRTOSZ	I	INTB
IPRINT		I	Print page counter initialization flag	/XCODES/(	168)	OUT	M	IPRINT
						TEST	I	IPRINT
						TOPM	O	IPRINT
ISTART		M	Initialization and divergance flag	/XCODES/(	147)	AST3	O	ISTART
						BLGCON	O	ISTART
						BLYNE	O	ISTART
						FNTG	I	ISTART
						MODELA	O	ISTART
						PROPIN	O	ISTART
						REU3	I	ISTART
						TEST	M	ISTART
						TOPM	M	ISTART
ITCT		M	Iteration counter	/XCODES/(	148)	BNTG	I	ITCT
						OUT	I	ITCT
						TEST	M	ITCT
						TOPM	M	ITCT
ITER		M	Trajectory pass indicator. ITER = 1, CONSTRAINTS = 2, OPTIMIZATION = 3, SOLUTION	/XCODES/(	149)	AST3	I	ITER
						FNTG	I	ITER
						GETIT	I	ITER
						MODELA	I	ITER
						OUT	I	ITER
						PAYO2	M	ITER
						PROPIN	I	ITER
						TEST	M	ITER
						TOPM	M	ITER
I2OP		M	First optimization pass flag sets $d\Phi$ = DPAY. Also used to indicate payoff degradation due to restoration of constraints	/XCODES/(	136)	PAYO2	M	I2OP
						TEST	M	I2OP
						TOPM	O	I2OP
KOP		M	Counts number of times constraint misses are halved down because of divergance problems	/XCODES/(	154)	TEST	M	KOP
						TOPM	I	KOP

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NCN		M	Number of elements in $d\Psi$	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BST03	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAYO2	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BST03	I	NCNST
						CON3	I	NCNST
						PAYO2	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST
NICNB		I	Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
						ADIC3A	I	NICNB
						BNTG	I	NICNB
						REU3	I	NICNB
						SDINP	M	NICNB
						TEST	I	NICNB
						TRAN3	I	NICNB
NITER		I	Maximum number of steepest descent iterations	/STS /(	31)	SDINP	I	NITER
						TEST	I	NITER
NOP		M	Counts number of times payoff is scaled down due to divergance problems	/XCODES/(	163)	TEST	M	NOP
						TOPM	I	NOP
NPARA		I	Number of adjustable parameters in trajectory problem	/PARAM /(	13)	ADJUST	I	NPARA
						BNTG	I	NPARA
						FNTG	I	NPARA
						MTX3A	I	NPARA
						PAYO2	I	NPARA
						PRMSET	I	NPARA
						SDINP	M	NPARA
						STAU	I	NPARA
						TEST	I	NPARA
						TOPM	O	NPARA
NPH		I	Number of phases in trajectory	/XCODES/(	164)	BNTG	I	NPH
						FNTG	O	NPH
						PRMSET	I	NPH
						SDINP	M	NPH
						TEST	I	NPH
						TOPM	I	NPH
NSAB		I	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB

710

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NST		I	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST
PMIN		I	Minimum payoff improvement	/STS /(	2)	PAYO2	I	PMIN
						SDINP	I	PMIN
						TEST	I	PMIN
PSISQ	$(dP)^2$	M	Metric of control and parameter changes [sd]	/GENF /(	302)	PAYO2	M	DPSQ
						TEST	M	PSISQ
						TRTOSZ	I	DPSQ
QV		M	Partial of dynamic pressure wrt velocity	/GENF /(	527)	EQUA3	M	QV
						TEST	M	QV
						VT	I	QV
SIG		I	Payoff sign. SIG < 0: Payoff to be minimized, SIG > 0: Payoff to be maximized.	/GLOBAL/(	65)	PAYO2	I	SIG
						SDINP	M	SIG
						TEST	I	SIG
						TRAN3	I	SIG
SUMSQ		M	Sum of squares of constraint misses (BCON) divided by tolerances on trial trajectory	/GENF /(	316)	TEST	M	SUMSQ
SVAR	$y\|_{t=0}$	I	Array of state values at initial problem time [sd]	/GENF /(	79)	ADJUST	O	SVAR
						BNTG	I	SVAR
						FNTG	I	SVAR
						PRMSET	M	SVAR
						REU3	I	SVAR
						SDINP	M	SVAR
						TEST	I	SVAR
						TOPM	I	SVAR
						TRTOSZ	I	SVAR
SVSQ		M	Same as sumsq but saved for nominal trajectory [sd]	/GENF /(	317)	TEST	M	SVSQ
TOL		I	Tolerence on constraint misses [sd]	/GENF /(	70)	SDINP	M	TOL
						SUMS	I	TOL
						TEST	I	TOL
TPH		I	Array of phase end times on trial trajectory [sd]	/GENF /(	351)	FNTG	O	TPH
						TEST	I	TPH
TPH1		O	Phase end times for nominal trajectory	/GENF /(	413)	BNTG	I	TPH1
						GETIT	I	TPH1
						SDINP	O	TPH1
						TEST	O	TPH1
						TOPM	I	TPH1
TST		I	Array of arc end times on trial trajectory [sd]	/GENF /(	331)	ADICB3	I	TST
						BNTG	I	TST
						FNTG	O	TST
						TEST	I	TST
TST1		O	Arc end times for nominal trajectory	/GENF /(	433)	BNTG	I	TST1
						GETIT	I	TST1
						PROPIN	I	TST1
						SDINP	O	TST1
						TEST	O	TST1
						TOPM	I	TST1
						TRAN3	I	TST1
						TRTOSZ	I	TST1
VARQ	⟨VARQ⟩	I	Desired constraint values [sd]	/GENF /(	61)	CON3	I	VARQ
						SDINP	M	VARQ
						TEST	I	VARQ
WORK		M	Working array, contains TOPEN1, TOPEN2, and PHIWT	/STS /(	3)	ADEQ3A	I	WORK
						FNTG	I	WORK
						MODELB	O	WORK
						MTX3A	I	WORK
						SDINP	M	WORK
						TEST	M	WORK

30 OCT 72 G.01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE		SUBROUTINF USAGE		
				BLOCK	LOC	SUBR	CODE	VAR
.UN06.		0	File of all output data	/.UN06 /(≑	)	BLICO	0	.UN06.
						BNDRYC	0	.UN06.
						CRASH	0	.UN06.
						FRENCH	0	.UN06.
						FXDAT	0	UN06.
						GEINP	0	.UN06.
						HUNT	0	.UN06.
						INEDIT	0	.UN06.
						ITERB	0	.UN06.
						MODELA	0	.UN06.
						MOMJ	0	.UN06.
						MPSI	0	UN06.
						OUT	0	.UN06.
						PAYO2	0	.UN06
						PRINT	0	.UN06.
						PRINTV	0	.UN06.
						PRINTW	0	.UN06.
						PRITEQ	0	.UN06.
						PRITVA	0	.UN06
						PROPIN	0	.UN06.
						PROTHR	0	.UN06.
						PRWTSM	0	.UN06.
						RANGE	0	.UN06.
						S	0	.UN06.
						SDINP	0	.UN06.
						SIZE	0	.UN06.
						SIZIN	0	.UN06.
						SIZOUT	0	.UN06.
						SOLVE	0	.UN06.
						SPLICO	0	.UN06
						SPLIZ	0	.UN06.
						SPLYNE	0	.UN06.
						SSSP	0	.UN06.
						STAU	0	.UN06.
						STPIT	0	.UN06.
						SUMOUT	0	.UN06.
						TABIN	0	.UN06.
						TEST	0	UN06.
						VEHDF	0	.UN06.
						WTSCH	0	.UN06.
						WTVOL	0	.UN06.

30 OCT 72 G 01-46

```
      PROGRAM  TEST                                                    TEST
C                                                                      COMN
C      CONSTRAINT MISS  AND  PERFORMANCE IMPROVEMENT EVALUATION        COMN
C                                                                      COMN
C                                                                      COMN
      COMMON/GENF/                                                     GENF
     *OMG(20)    ,OMGP(20,2),VARD(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP       , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE         ,FP        ,FPOLD     ,FPO       ,MACHR     ,MACHV     , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                 GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED     ,IRATED    ,                                           GENF
     *P1         ,P2        ,P3        ,XK1       ,XK2       ,XK3       , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,               MACH, MACHR,   GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                 FRAT
      DIMENSION TPH1(10),TST1(10)                                      GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                               GENF
      COMMON /XCODES/                                                  XCODES
     *ITQ    (9),ICOR  (20),ITI       ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN      ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3       XCODES
      COMMON/GLOBAL/                                                   GLOBAL
     *GR         ,ER        ,OMGZ      ,XLAMRF    ,YMURF     ,LUM        GLOBAL
     *,JJOP(10)  ,IFATAL    ,NARC      ,NBRAN     ,NFARC     ,ID(4)      GLOBAL
     *,KTAB(20),ITAB(20)  ,SIG    ,MAXTAB                                GLOBAL
     *,GM        ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)      GLOBAL
     *,ITPSO  ,KSOL     ,KGLOBL(8)                                       RETAP
      COMMON/STS/                                                      STS
     *DPAY       ,PMIN      ,WORK (20),NWDS      ,IPC    (7),NITER    , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)          STS
      EQUIVALENCE(DPSQ,PSISQ)                                          TEST
      COMMON/PARAM/                                                    PARAM
     *IPOINT(12),NPARA,NPA ,SPARA(9,12),WTPD  (9),WTP   (12),          PARAM
     *SPARB(9,12) , PARA(12),DPAR(12) ,S2INV(9,9)                      PARAM
     *,DELP(9)                                                         PARAM
C                                                                      COMN
C         I       TEST  FLAGS                                          COMN
C         I-A     TEST  FOR  DIVERGED  CONTROL  ERROR  GO TO IV-E OR VI-B COMN
      IF(IPRINT.NE.1) GO TO(150,250),ITER                              TEST
C         I-B        FIRST NOMINAL  TRAJECTORY TEST  --NOT  GO TO IV   COMN
      IF(ISTART GT 1) GO TO 80                                         TEST
C         II          FIRST NOMINAL  LOGIC                             COMN
      IF(NCNST EQ.0)  GO TO  50                                        TEST
C         II-A       IF  NO  CONSTRAINTS  GO TO III                    COMN
      IF(INTB-1) 10,30,30                                              TEST
C         II-B       BRANCHING  OR INTERMEDIATE  CONSTRAINT GO TO II-D COMN
   10 CALL  CONIN(1,NCNST)                                             TEST
```

713

```
C         II-C    EVALUATE  ALL  TERMINAL  CONSTRAINT  MISSES (NOMINAL)   COMN
C                 GO  TO   VIII                                           COMN
     20 SVSQ = SUMS(DCON)                                                 TEST
        GO  TO  40                                                        TEST
     30 CONTINUE                                                          TEST
        CALL  CONIN (NICNB+1,NCNST)                                       TEST
        GO  TO 20                                                         TEST
     40 CONTINUE                                                          TEST
        NCN = NCNST                                                       TEST
        GO TO  380                                                        TEST
C          III     NO  CONSTRAINTS  (NOMINAL)                             COMN
     50 ITER =2                                                           TEST
C         III-A    EVALUATE   PAYOFF  FUNCTION  AND                       COMN
C                DETERMINE INITIAL  STEP  SIZE                            COMN
C                SET FIRST OPT. PASS FLAP I2OP=1 , GO TO VIII             COMN
         CALL CONIN(1,NCNST)                                              TEST
        IF(VARQ(NCN).EQ.0.) GO TO 60                                      TEST
        QV=VARQ(NCN)-ACON(NCN)                                            TEST
        IF(ABS(QV)/4.0.LT.ABS(DPAY)) GO TO 60                             TEST
        DCON(NCN)=QV/4.                                                   TEST
        GO TO 70                                                          TEST
     60 CONTINUE                                                          TEST
        DCON(1)= DPAY                                                     TEST
     70 CONTINUE                                                          TEST
        I2OP =1                                                           TEST
        NCN =1                                                            TEST
        GO TO 380                                                         TEST
C          IV        TRIAL  TRAJECTORY    LOGIC                           COMN
C          IV-A      IF NO CONSTRAINTS    GO  TO VII                      COMN
     80 IF(NCNST.EQ.0) GO TO 330                                          TEST
C          IV-B      EVALUATE  TERMINAL  CONSTRAINT MISSES                COMN
C                 AND  THE  SUM OF THEIR  SQUARES                         COMN
C                 IF  OPTIMIZATION  PASS  GO  TO VI                       COMN
        IF(INTB-1) 90,110,110                                             TEST
     90 CALL  CON (1,NCNST)                                               TEST
    100 SUMSQ = SUMS(BCON)                                                TEST
        IF(ITER.EQ.2) GO TO 240                                           TEST
        GO TO 120                                                         TEST
    110 CALL  CON (NICNB+1,NCNST)                                         TEST
        GO TO 100                                                         TEST
C          IV-C    THIS  IS  A   GUIDANCE  PASS ( SATISFY CONSTRAINTS ONLY) COMN
C                    COMPARE  CONSTRAINT  MISSES WITH TOLERANCES          COMN
C          IF  ALL  LESS GO TO V    ,ELSE   GO  TO  IV-D                  COMN
C                                                                         COMN
    120 DO 130 I=1,NCNST                                                  TEST
        IF(ABS(BCON(I)).GT.TOL(I)) GO TO 140                              TEST
    130 CONTINUE                                                          TEST
        GO TO 190                                                         TEST
C          IV-D    IS  CURRENT SUMSQ LESS THAN ON  PREVIOUS  ITERATION    COMN
C              YES  MEANS  CONVERGING  TO TO IV-F ELSE  GO TO IV-E        COMN
    140 IF( SUMSQ.LT.SVSQ)  GO  TO  170                                   TEST
C         IV-E    NON-CONVERGANT  GUIDANCE  PASS , HALVE  DOWN            COMN
C                 CONSTRAINT  CORRECTIONS ( UP TO 5 TIMES) AND            COMN
C                 RETURN FOR  NEW  TRIAL  TRAJECTORY                      COMN
    150 KOP=KOP +1                                                        TEST
        IF(KOP.GE.5) GO TO 450                                            TEST
        DO 160  I=1,NCNST                                                 TEST
    160 DCON(I) =DCON(I)*.5                                               TEST
        WRITE(6,520)                                                      OS
        RETURN                                                            TEST
    170 DO 180  I=1,NCNST                                                 TEST
C         IV-F    SET UP  NOMINAL  CONSTRAINT MISSES FOR NEXT ITERATION   COMN
C                 SAVE  SUMSQ  GO TO VIII                                 COMN
        BCON(I)=BCON(I)                                                   TEST
    180 ACON(I)=BCON(I)                                                   TEST
        NCN =NCNST                                                        TEST
        ITER =1                                                           TEST
        SVSQ = SUMSQ                                                      TEST
```

20 OCT 72 6.01-46

114

```
      GO TO 374                                                         TEST
C                                                                       COMN
C       V         SWITCH  TO  OPTIMIZATION  PASS                        COMN
C                 SET UP  NOMINAL  CONSTRAINT MISSES FOR NEXT  ITERATION COMN
C                    GO  TO  VIII                                       COMN
  190 DO 200  I=1,NCNST                                                 TEST
      DCON(I)=BCON(I)                                                   DS
      ACON(I)=BCON(I)                                                   DS
  200 CONTINUE                                                          TEST
C       V-A    EXPAND  CONSTRAINT VECTOR BY ONE  TO  INCLUDE PAYOFF     COMN
      NCN =NCNST +1                                                     TEST
C       V-B    DETERMINE  FIRST  PAYOFF  IMPROVEMENT                    COMN
      ACON(NCN) =BCON(NCN)                                              TEST
      IF(I2OP EQ 3) GO TO 230                                           TEST
      IF(VARQ(NCN) EQ 0.) GO TO 210                                     TEST
      QV=VARQ(NCN)-BCON(NCN)                                            TEST
      IF(SIGN(1.,QV) NE.SIG) GO TO 210                                  TEST
      IF(ABS(QV)/4 0.LT.ABS(OPAY)) GO TO 210                            TEST
      DCON(NCN)=QV/4.                                                   TEST
      GO TO 220                                                         TEST
  210 CONTINUE                                                          TEST
      DCON(NCN)= OPAY                                                   TEST
  220 CONTINUE                                                          TEST
C                                                                       COMN
C       V-C       SET  PASS TYPE FLAG  AND  FIRST OPT. FLAG             COMN
C                                                                       COMN
C                  GO  TO  VIII                                         COMN
      ITER =2                                                           TEST
      I2OP = 1                                                          TEST
      SVSQ = SUMSQ                                                      TEST
      GO  TO  370                                                       TEST
  230 ITER=2                                                            TEST
      SVSQ= SUMSQ                                                       TEST
      GO TO 370                                                         TEST
C       VI                                                              COMN
C     OPTIMIZATION  PASS CHECKING                                       TEST
  240 CONTINUE                                                          TEST
C       VI -A  IS PAYOFF  IMPROVING , YES  GO TO VI-C,NO  GO TO VI-B    COMN
      IF(SIG*(BCON(NCN)- ACON(NCN)))250,250,260                         TEST
C       VI-B      SCALE  DOWN  ASKED  FOR PAYOFF  IMPROVEMENT           COMN
  250 NOP =NOP+1                                                        TEST
      IF(KOP+NOP LE 7) GO TO 255                                        AAA
      IF(ITCT GT 8) GO TO 470                                           AAA
      IF(ABS(DCON(NCN)).LT.*.*PMIN) GO TO 470                           AAA
  255 CONTINUE                                                          AAA
      DCON(NCN)=DCON(NCN)*.707107                                       TEST
      WRITE(6,530)                                                      DS
      RETURN                                                            TEST
  260 CONTINUE                                                          TEST
      IF(I2OP EQ.3) GO TO 280                                           TEST
C       VI-C     ***                                                    COMN
C    CHECK FOR  CONSTRAINT  VIOLATIONS                                  TEST
C           IF NONE  GO  TO VI-D ELSE   GO TO  VI-E                     COMN
      DO 270  I=1,NCNST                                                 TEST
      IF(ABS(BCON(I)) GT TOL(I)) GO TO 320                              TEST
  270 CONTINUE                                                          TEST
      GO TO 300                                                         TEST
  280 DO 290 I=1,NCNST                                                  TEST
      IF(ABS(BCON(I)) GT.TOL(I)/2.) GO TO 320                           TEST
  290 CONTINUE                                                          TEST
C       VI-D       SET  UP  NOMINAL CONSTRAINT  MISSES  AND PAYOFF      COMN
C                  FOR  NEXT  ITERATION    GO  TO  VIII                 COMN
  300 CONTINUE                                                          TEST
      DO 310 I=1,NCNST                                                  TEST
      ACON(I)=BCON(I)                                                   DS
      DCON(I)=BCON(I)                                                   DS
  310 CONTINUE                                                          TEST
      ITER =2                                                           TEST
      ACON(NCN)=BCON(NCN)                                               TEST
      NCN = NCNST+1                                                     TEST
```

```
      GO TO  450                                                        TEST
C      VI F     CONSTRAINTS  VIOLATED ON OPTIMIZATION PASS              COMN
C      SCALE  DOWN   PAYOFF  IMPROVEMENT   RETURN                       COMN
  320 KOP = KOP +1                                                      TEST
      IF(KOP GE 5) GO TO  490                                           TEST
      IF(KOP+NOP.GE 7) GO TO  490                                       TEST
      DCON(NCN)=DCON(NCN)* 707107                                       TEST
      WRITE(6,540)                                                      OS
      RETURN                                                            TEST
C           VIII  **                                                    COMN
      OPTIMIZATION   WITHOUT  CONSTRAINTS                               TEST
C           VII-A     EVALUATE PAYOFF   AND  DETERMINE IF IT HAS        COMN
C                  IMPROVED  IF NOT  SCALE DOWN  ,OK  GO  TO  VII-B     COMN
  330 CONTINUE                                                          TEST
      CALL CON(1,0)                                                     TEST
      IF(SIG*(BCON(1)-ACON(1))) 340,350,350                             TEST
  340 NOP= NOP+1                                                        TEST
      IF(NOP.GE.5) GO TO 470                                            TEST
      DCON(1  )=DCON(1  )* 707107                                       TEST
      WRITE(6,530)                                                      OS
      RETURN                                                            TEST
  350 CONTINUE                                                          TEST
C        VII-B      SET UP  NOMINAL PAYOFF FOR  NEXT ITERATION          COMN
      ACON(1)=BCON(1)                                                   TEST
      ITER =2                                                           TEST
      NCN =1                                                            TEST
C                                                                       COMN
C       VIII      SET  UP   DATA  FOR    NEXT  ITERATION                COMN
C            SAVE   ARC AND PHASE TIMES    ,SCALE  CONTROL METRIC (PSISQ)  COMN
C            SAVE  NOMINAL  PARAMETER  VALUES                           COMN
  360 CONTINUE                                                          TEST
  370 PSISQ=PSISQ/(2 0**(KOP+NOP))                                      TEST
      WORK(5)=WORK(5)*(.707107)**(KOP+NOP)                              TEST
  380 CONTINUE                                                          TEST
      IF(NPARA GT.0) CALL PRMSET(0 )                                    TEST
      KOP = 0                                                           OS
      NOP = 0                                                           OS
      IF(NST EQ 1) GO TO 400                                            TEST
      DO 390 I=2,NST                                                    TEST
      DIS1(I)=TST(I)                                                    TEST
  390 TST1(I)=TST(I-1)                                                  TEST
  400 TST1(1) =SVAR(1)                                                  TEST
      DIS1(1)=TST(1)                                                    TEST
      IF(NPH.EQ 1) GO TO 420                                            TEST
      DO 410 I=2,NPH                                                    TEST
      DIP1(I)=TPH(I)                                                    TEST
  410 TPH1(I)=TPH(I-1)                                                  TEST
  420 TPH1(1)=SVAR(1)                                                   TEST
      DIP1(1)=TPH(1)                                                    TEST
      IF(INTB NE.2) GO TO 430                                           TEST
      I=NSB+NSAB+1                                                      TEST
      TST1(I)=TST(NSB)                                                  TEST
      TPH1(NPHP)=TPH(NPHB)                                              TEST
C       IX       TEST IF  ITERATIONS  ARE EXCEEDED  IF YES  CALL FOR    COMN
C                SOLUTION  TRAJECTORY  ELSE  GO  TO  X                  COMN
  430 CONTINUE                                                          TEST
      ITCT =ITCT +1                                                     TEST
      IF(ITCT.GT NITER) GO TO 440                                       TEST
C       X       CALCULATE   ADJOINT  INITIAL  CONDITIONS                COMN
      CALL  ADIC                                                        TEST
      RETURN                                                            TEST
  440 ITER=3                                                            TEST
      RETURN                                                            TEST
C   ERROR  RETURNS                                                      TEST
  450 WRITE(6,460)                                                      OS
  460 FORMAT(33HO UNABLE TO SATISFY CONSTRAINTS            )            OS
      GO TO 510                                                         TEST
  470 WRITE(6,480) ACON(NCN),BCON(NCN)                                  OS
  480 FORMAT (18H  NOMINAL PAYOFF= E20.10,17H   TRIAL PAYOFF= E20.10/,  TEST
```

20 OCT 72 G.01-46

```
     * 55H   NO FURTHER OPTIMIZATION TRIALS WILL BE ATTEMPTED          ) P014
      GO TO 510                                                         TEST
  490 WRITE(6,500)                                                      OS
  500 FORMAT (48H   CONSTRAINTS VIOLATED DURING OPTIMIZATION PASS)      TEST
C     XI            CONVERGANCE DIFFICULTY  RETURN                      COMN
C             DISREGARD  CURRENT TRIAL  AND  USE  LAST  ITERATION       COMN
C               AS  SOLUTION  TRAJECTORY                                COMN
  510 ISTART =5                                                         TEST
      IF(NPARA.GT 0) CALL PRMSET(1)                                     TEST
      RETURN                                                            TEST
  520 FORMAT (53H   TRIAL CONSTRAINT MISSES DID NOT  DECREASE 1/2 DOWN) TEST
  530 FORMAT (45H   TRIAL PAYOFF  DID NOT IMPROVE    1/2  DOWN)         TEST
  540 FORMAT (50H   TRIAL  CONSTRAINT TOLERANCES EXCEEDED  1/2 DOWN)    TEST
      END                                                               TEST
```

510

FUNCTION

TOLPSI

FUNCTION TØLPSI

Entry Point SETTOL

Purpose

TØLPSI supplies a preset constraint tolerance to SDINP during the boundary condition scan.

SETTØL is an entry point which permits update of preset constraint tolerances.

```
      FUNCTION TOLPSI( I)                                                TOLPSI
C                                                                        COMN
C        PRESET  DATA  FOR  CONSTRAINT MISS  TOLERANCES                  COMN
C                                                                        COMN
      DIMENSION TAL(36),I(1),IT(1)                                       RETAP
      EQUIVALENCE (IT,TAL)                                               RETAP
C        SETS   CONSTRAINT  MISS  TOLERANCES                             RETAP
C                     TAU        V          GAM       ALT       MASS     RETAP
      DATA  TAL /      0.   ,   20.    ,  .01    ,  2000.   ,  .2   ,    RETAP
C                     PSI       RHO        MU        TIME      HEATLD    RETAP
     *                .07   ,   001    ,  001    ,  .2      ,  10.  ,    RETAP
C                     SQUIG     VI         GAMI      PSII      MUI       RETAP
     *                20    ,   20     ,  .01    ,  .07     , .001  ,    RETAP
C                     SEMI LAT  ECC         INCL     ARGP      ASCNOD    RETAP
     *               10000. , .00025   ,  .01    , .01      , .001  ,    RETAP
C                    SEMI MAJ  APOGEE    PERGEE     ANOMLY    CAP X      RETAP
     *               10000  , 10000    , 10000.  ,  .01     , 10000. ,   RETAP
C                    CAPY      ASSYM     ENERGY    MOMENTUM   SD         RETAP
     *               10000. ,  .01     , 1 E5    , 1.E5     ,  6000. ,   RETAP
C                    SC        STOT       Q         HTDOT     REY        RETAP
     *               6000.  ,  6000.   , .5      , .1       , 200.  ,    RETAP
C                   FUEL WT                                              RETAP
     *        30./                                                       APR72
      TOLPSI = TAL(I)                                                    TOLPSI
      RETURN                                                             TOLPSI
C          OVERIDE  PRESET  DATA  WITH INPUT                             RETAP
      ENTRY SETTOL                                                       RETAP
      DO 10 J=1,36                                                       RETAP
      IF(I(J) NE.0) IT(J) = I(J)                                         RETAP
   10 CONTINUE                                                           RETAP
      SETTOL = 0.                                                        RETAP
      RETURN                                                             RETAP
      END                                                                TOLPSI
```

20 OCT 72 G 01-46

SUBRØUTINE
TØL3

FUNCTION TOL 3

Entry Points. TOLF, TOLPH

Purpose

Function TOL 3 sets up data for cut-off refinement during the forward trajectory.

Description

Entry points TOLF and TOLPH are called from FNTG during arc and phase cut-off refinement calculations.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
FP		I	Current value of cut-off function - non-linear only	/GENF /(	521)	DTF3	I	FP
						STP3	I	FP
						TOL3	I	FP
						VREF3	I	FP
FPOLD		0	Value of non-linear cut-off function at prior compute interval	/GENF /(	522)	DTF3	I	FPOLD
						STP3	0	FPOLD
						TOL3	0	FPOLD
						VREF3	I	FPOLD
IVAR		I	Cut-off variable option indicator	/XCODES/(	150)	FNTG	M	IVAR
						STP3	0	IVAR
						TOL3	I	IVAR
JPS		I	Absolute value of phase cut-off option code	/XCODES/(	152)	ADID3A	I	JPS
						BNTG	M	JPS
						FNTG	M	JPS
						STP3	I	JPS
						TOL3	I	JPS
JS		I	Absolute value of arc cut-off option code	/XCODES/(	153)	ADICB3	M	JS
						ADIC3A	I	JS
						ADID3A	I	JS
						BNTG	M	JS
						FNTG	M	JS
						PROPB	I	JS
						PROPIN	I	JS
						STP3	I	JS
						TOL3	I	JS

```
      FUNCTION TOL3(C)                                                    TOL3
C                                                                         COMN
C        SET -UP  DATA  FOR  CUT-OFF  REFINEMENT                          COMN
C                                                                         COMN
      COMMON/GENF/                                                        GENF
     *OMG(20)    ,OMGP(20,2),VARD(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)   , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP       , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS        , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS        , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ     , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)     , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W         , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,DMP       , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)   , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV     , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA     , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF      , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA     , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE     , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID       , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                     GENF
      COMMON / GENF  /                                                    GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP       , GENF
     *FRATED    ,IRATED     ,                                             GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3       , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D      , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V      , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P      , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O      , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M      , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8) , GENF
      REAL  LIFTR , LIFT  , LIFTA,LIFTM ,                MACH, MACHR,     GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                    FRAT
      DIMENSION TPH1(10),TST1(10)                                         GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                  GENF
      COMMON /XCODES/                                                     XCODES
     *ITQ    (9),ICOR  (20),ITI        ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB     , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND       , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART    , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS        , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE     , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N         , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN      ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON/STATE3/                                                      STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10)   , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2    , STATE3D
     *SVBV    (9),OMEGA     ,OMEGAZ    ,                                  STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV       , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG       , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR       , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP       , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO       , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                              STATE3D
      REAL MDM , MDV, MDR                                                 STATE3D
      COMMON/STATE3/                                                      STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                       STATE3D
      ENTRY TOLF                                                          TOL3
      IF(IVAR GT 7) GO TO 10                                              TOL3   10
      TOL3=1                                                              TOL3
      RETURN                                                              TOL3
   10 MO=0                                                                TOL3
      FPOLD= FP                                                           TOL3
      CALL PDBC(JS,FP,S,SD,0,MO)                                          TOL3
      TOL3 =1.                                                            TOL3
      RETURN                                                              TOL3
      ENTRY TOLPH                                                         TOL3
      IF(IVAR.GT.7) GO TO 20                                              TOL3   20
      TOL3=1                                                              TOL3
      RETURN                                                              TOL3
```

```
   20 FPOLD = FP                                                      TOL3
      MO=0                                                            TOL3
      CALL PDBC(JPS,FP,S,SD,0,MO)                                     TOL3
      TOL3 =1.                                                        TOL3
      RETURN                                                          TOL3
      END                                                             TOL3
```

20 OCT 72 G 01-46

SUBRØUTINE
TRAN3

TRAN 3

Purpose

TRAN3 computes transformed adjoints to serve as approximations for Euler-Lagrange Multipliers. This approximation is used as a starting solution for the quasi-linearization module.

Description

The equations for TRAN3 are described in Section 14 of Volume I.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
A	A	I	Control integral matrix	/GENF /(	109)	ADEQ3A	O	A
						ADICB3	M	A
						BGET3	O	A
						BNTG	I	A
						BST03	I	A
						MTX3A	I	A
						PAY02	I	A
						SDINP	I	A
						TRAN3	I	A
COTI		M	Temp storage for a matrix also called B matrix	/GENF /(	208)	ADICB3	M	COTI
						MTX3A	M	B
						TRAN3	M	COTI
DIS1		I	Arc initial times for nominal trajectory [sd]	/GENF /(	473)	GETIT	I	DIS1
						SDINP	M	DIS1
						TEST	O	DIS1
						TOPM	I	DIS1
						TRAN3	I	DIS1
FTIME		I	Time at which trajectory data set is stored (SEC)	/RETREV/(	1)	AGETB3	O	FTIME
						AST3	O	FTIME
						GETIT	I	FTIME
						TRAN3	I	FTIME
INTB		I	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
						BNTG	I	INTB
						ENVPRM	I	INTB
						FNTG	I	INTB
						SDINP	M	INTB
						TEST	I	INTB
						TRAN3	I	INTB
						TRTOSZ	I	INTB
ISTN		I	Stored history data arc number	/XCODES/(	169)	AGETB3	O	ISTN
						AST3	O	ISTN
						GETIT	I	ISTN
						TRAN3	I	ISTN
NCN		I	Number of elements in d*	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BST03	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAY02	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NCNST	n	I	Number of problem constraints	/XCODES/(	132)	BGET3	I	NCNST
						BST03	I	NCNST
						CON3	I	NCNST
						PAY02	I	NCNST
						SDINP	M	NCNST
						SUMS	I	NCNST
						TEST	I	NCNST
						TOPM	I	NCNST
						TRAN3	I	NCNST

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BSTD3	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
NEQF		O	Number of equations to be integrated on forward trajectory	/XCODES/(	185)	REU3	I	NEQF
						RKTA3A	I	NN
						SDINP	O	NEQF
						STAU	I	NEQF
						TOPM	O	NEQF
						TRAN3	O	NEQF
NICNB		I	Number of constraints at intermediate constraint point or at end of first branch	/XCODES/(	135)	ADICB3	I	NICNB
						ADIC3A	I	NICNB
						BNTG	I	NICNB
						REU3	I	NICNB
						SDINP	M	NICNB
						TEST	I	NICNB
						TRAN3	I	NICNB
NSAB		I	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB
NST		I	Number of arcs in trajectory	/XCODES/(	166)	BNTG	I	NST
						FNTG	O	NST
						PROPB	I	NST
						SDINP	I	NS
						SDINP	M	NST
						TEST	I	NST
						TOPM	I	NST
						TRAN3	I	NST
SIG		I	Payoff sign SIG < 0. Payoff to be minimized, SIG > 0 Payoff to be maximized.	/GLOBAL/(	65)	PAYOZ	I	SIG
						SDINP	M	SIG
						TEST	I	SIG
						TRAN3	I	SIG
TR		M	Vector modifier of impulse response function in control calculation	/GENF /(	322)	MTX3A	M	TR
						TRAN3	M	TR
TST1		I	Arc end times for nominal trajectory	/GENF /(	433)	BNTG	I	TST1
						GETIT	I	TST1
						PROPIN	I	TST1
						SDINP	O	TST1
						TEST	O	TST1
						TOPM	I	TST1
						TRAN3	I	TST1
						TRTOSZ	I	TST1

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
VARL		M	Array of variables for adjoint integration	/STATE3/(	29)	ADEQ3A	I	VARL
						ADICB3	O	VARL
						ADIC3A	O	VARL
						PROPIN	I	ZZ
						RKTA3A	M	F
						RKTB3A	M	Y
						STVRL3	O	VARL
						TRAN3	M	VARL
XL	$\lambda^{\Psi_i \Omega_J}$	M	Matrix of adjoint variables	/STATE3/(	246)	ADEQ3A	M	XL
						ADICB3	M	XL
						ADIC3A	M	XL
						ADID3A	M	XL
						AST3	M	XL
						BGET3	O	XL
						BSTO3	I	XL
						MTX3A	I	XL
						OUT	I	XL
						STAU	M	XL
						STVRL3	I	XL
						TRAN3	M	XL
XLAMA	$\Lambda^{\Psi_i \Omega_J}$	M	Impulse response function column vector associated with angle of attack	/AECO3 /(	16)	ADEQ3A	M	XLAMA
						ADIC3A	O	XLAMA
						AST3	O	XLAMA
						BGET3	O	XLAMA
						BSTO3	M	XLAMA
						MTX3A	I	XLAMA
						TRAN3	M	XLAMA
XLAMP	$\Lambda^{\Psi_i \Omega_J}$	M	Impulse response function column vector associated with bank angle	/AECO3 /(	25)	ADEQ3A	M	XLAMP
						ADIC3A	O	XLAMP
						AST3	O	XLAMP
						BGET3	O	XLAMP
						BSTO3	M	XLAMP
						MTX3A	I	XLAMP
						TRAN3	M	XLAMP

```
      PROGRAM TRAN3                                                     TRAN3
C                                                                       COMN
C        COMPUTED  TRANSFORMED ADJOINTS  TO  SERVE  AS                  COMN
C        APPROXIMATION FOR EULER  LAGRANGE  MULTIPLIERS                 COMN
      COMMON/STATE3/                                                    STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10) , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO    ,OCORO2   , STATE3D
     *SVBV   (9),OMEGA      ,OMEGA2     ,                                STATE3D
     *VDV        ,GDV        ,RDV        ,MDV        ,PDV       ,ODV      , STATE3D
     *UDV        ,VDG        ,GDG        ,RDG        ,PDG       ,ODG      , STATE3D
     *UDG        ,VDR        ,GDR        ,MDR        ,PDR       ,ODR      , STATE3D
     *UDR        ,VDM        ,GDM        ,MDM        ,PDM       ,VDP      , STATE3D
     *GDP        ,PDP        ,ODP        ,UDP        ,VDO       ,GDO      , STATE3D
     *PDO        ,UDO        ,HTDV       ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                               STATE3D
      COMMON/STATE3/                                                    STATE3D
     *SIN2RO     ,COS2RO     ,COS2GM                                    STATE3D
      COMMON/AECO3/                                                     AECO3
     *APHD       ,APHR       ,ALPHA      ,VDA        ,GDA       ,PDA      , AECO3
     *SINA       ,COSA       ,PHIO       ,PHID       ,PHI       ,SINPHI   , AECO3
     *COSPHI     ,GDPH       ,PDPH       ,XLAMA(9)   ,XLAMP(9)  ,CDO      , AECO3
     *CDOM       ,CLO        ,FK         ,XCGM       ,ZCGM      ,CLOM     , AECO3
     *CM         ,CMA        ,CMAM ,CMM  ,CMO        ,CMOM      ,FKM      , AECO3
     *CLAM       ,CL         ,CLA        ,CLM        ,                     AECO3
     *CD         ,CDA        ,CDM                                         AECO3
      COMMON/GLOBAL/                                                    GLOBAL
     *GR         ,ER         ,OMGZ       ,XLAMRF     ,YMURF     ,LUM        GLOBAL
     *,JJOP(10) ,IFATAL     ,NARC          ,NBRAN     ,NFARC     ,ID(4)     GLOBAL
     *,KTAB(20),ITAB(20)   ,SIG     ,MAXTAB                             GLOBAL
     *,GM        ,PSIRF,IPFLG1,      IPFLG2,IPFLG3,IPFLG4,INEQFL(20)      GLOBAL
     *,ITPSO ,KSOL     ,KGLOBL(8)                                        RETAP
      COMMON/STS/                                                       STS
     *DPAY       ,PMIN       ,WORK (20),NWDS       ,IPC    (7),NITER    , STS
     *MNGA(20,2),MNGP(20,2) ,AR(200),IAD(20) ,INP(20),ISV(20)           STS
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)    ,TOL(9)     ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)    ,BCON(9)    ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT         ,G          ,DPSQ       ,Q         ,QS       , GENF
     *R          ,RE         ,MACH       ,PA         ,RO        ,CS       , GENF
     *VNU        ,PAR        ,ROR        ,CSR        ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH     ,TIMES      ,TOP        ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),OIS(20)    ,DIP(20)    ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)   ,DIS1(20)   ,TIME      ,OMP      , GENF
     *TIMPA      ,LIFT       ,DRAG       ,TAX        ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP         ,FPOLD      ,FPD        ,MACHR     ,MACHV    , GENF
     *QR         ,QV         ,FVAC       ,LIFTV      ,                     GENF
     *LIFTR      ,LIFTA      ,            DRAGV      ,DRAGR     ,DRAGA    , GENF
     *            LIFTM      ,DBR        ,DB         ,ISP       ,ISPF     , GENF
     *                        ULFT       ,ULFTV      ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV      ,XMCGR      ,XMCGA      ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT         ,CALPHA     ,CDE        ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE      ,XCG        ,ZCG        ,XJ                    GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR        ,GH         ,GAMMAD     ,XKG       ,XKP      , GENF
     *FRATED   ,IRATED     ,                                             GENF
     *P1         ,P2         ,P3       ,XK1          ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T       ,XK3T       ,XK1D       ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A       ,XK3A       ,XK1V       ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G       ,XK3G       ,XK1P       ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R       ,XK3R       ,XK1O       ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U       ,XK3U       ,XK1M       ,XK2M      ,XK3M     , GENF
     *PV         ,PG         ,PP         ,PR         ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                MACH, MACHR,    GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR  (20),ITI         ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)             ,NCNST      ,NSB        ,NSAB      ,NICNB    , XCODES
     *IZOP       ,ICOP       ,IFAW       ,IFAR       ,IFB       ,IND      , XCODES
     *IOPEN      ,IPH        ,ISPH       ,ISST       ,IARC      ,ISTART   , XCODES
     *ITCT       ,ITER       ,IVAR       ,JK         ,JPS       ,JS       , XCODES
     *KOP        ,KPST       ,K          ,KST        ,NAD       ,NCASE    , XCODES
     *NCN        ,NEQB       ,NEQ        ,NOP        ,NPH       ,N        , XCODES
```

20 OCT 72 G 01-46

```
     *NST       ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB     , XCODES
     *IPHNB     ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L         , XCODES
     *IFOB      ,NB        ,LB        ,MB        ,NPHP      ,NPHB      , XCODES
     *NCTIN     ,NEQF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3        XCODES
      COMMON/RETREV/ FTIME,BTIME,MAXA(2),MAXB                            RETREV
      COMMON/RETREV/                                                     RETREV
     *NBUFA(2)  ,IBUF1     ,IBUF2     ,NBFA      ,NBFB      ,MIXA      , RETREV
     *MIXB      ,MXA       ,MXB       ,NPTA      ,NPTB      ,IBLKB     , RETREV
     *,NBUFB    ,IBUFB                                                   RETREV
      DIMENSION JJ(9)                                                    TRAN3
C                                                                        COMN
C       I     INITIALIZE   STORAGE  RETRIEVAL                            COMN
      CALL BGRA                                                          TRAN3
      CALL BGRB                                                          TRAN3
      NEQF=NEQ+5                                                         TRAN3
      REWIND  10                                                         TRAN3
C                                                                        COMN
C      II     IF   NO  CONSTRAINTS   GO  TO  XII                         COMN
      IF(NCNST.EQ 0) GO TO 230                                           TRAN3
C      III    GET  FIRST  A  MATRIX  AT  INITIAL TIME                    COMN
      CALL  BGET                                                         TRAN3
      NO=0                                                               TRAN3
C      IV     COMPRESS  A  MATRIX, STORE  IN  COTI  AND  THEN            COMN
C             INVERT  COTI                                               COMN
      DO 20  I=1,NCN                                                     TRAN3
      IF(A(I,I).EQ 0.) GO TO 10                                          TRAN3
      JJ(I)= 1                                                           TRAN3
      NO =NO +1                                                          TRAN3
      GO TO  20                                                          TRAN3
   10 JJ(I) =0                                                           TRAN3
   20 CONTINUE                                                           TRAN3
      IF(NO.EQ 1) GO TO 230                                              TRAN3
      IA = 0                                                             TRAN3
      DO 40 I=1,NCN                                                      TRAN3
      IF(JJ(I).EQ.0) GO TO 40                                            TRAN3
      IA=IA+1                                                            TRAN3
      IB= IA-1                                                           TRAN3
      DO 30 J=I,NCN                                                      TRAN3
      IF( JJ(J).EQ 0) GO TO 30                                           TRAN3
      IB =IB +1                                                          TRAN3
      COTI(IA,IB) = A(I,J)                                               TRAN3
   30 CONTINUE                                                           TRAN3
   40 CONTINUE                                                           TRAN3
      NM = NO -1                                                         TRAN3
      IF(NM GT.1) GO TO 50                                               TRAN3
      COTI(1,1)= 1 / COTI(1,1)                                           TRAN3
      GO TO 80                                                           TRAN3
   50 CONTINUE                                                           TRAN3
      CALL  SYMVRT( COTI,NM  ,IER)                                       TRAN3
      IF(IER NE 0) GO TO 270                                             TRAN3
      DO 70  I=1,NM                                                      TRAN3
      DO 60  J=I,NM                                                      TRAN3
   60 COTI(I,J) = COTI(J,I)                                              TRAN3
   70 CONTINUE                                                           TRAN3
   80 CONTINUE                                                           TRAN3
C                                                                        COMN
C       V     COMPUTE   TRANSFORMATION  MULTIPLIERS                      COMN
      DO 100  J=1,NM                                                     TRAN3
      TR(J)= 0                                                           TRAN3
      DO 90 I= 1,NM                                                      TRAN3
   90 TR(J)= TR(J) + COTI(I,NO)*COTI(I,J)                                TRAN3
  100 CONTINUE                                                           TRAN3
      J=0                                                                TRAN3
C                                                                        COMN
C       VI     ELIMINATE  NULL MEMBERS ACCORPING TO HOW A WAS COMPRESSED COMN
      DO 120 I=1,NCNST                                                   TRAN3
      IF(JJ(I) EQ.0) GO TO 110                                           TRAN3
      J=J+1                                                              TRAN3
      VARL(I)=TR(J)                                                      TRAN3
      GO TO 120                                                          TRAN3
  110 VARL(I)=0                                                          TRAN3
```

20 OCT 72 G 01-46

```
      120 CONTINUE                                                       TRAN3
          DO 130 I=1,NCNST                                               TRAN3
      130 TR(I)= VARL(I)                                                 TRAN3
          CALL BGRB                                                      TRAN3
C                                                                        COMN
C          VIII TRANSFORM  ENTIRE  TIME  HISTORY                         COMN
C                                                                        COMN
C          VII    INITIALIZE  RETRIEVAL  AGAIN                           COMN
          DO 210  KJPOS =1,2000                                          TRAN3
C          VIII-A   GET  ADJOINT MATRIX                                  COMN
          CALL  AGETA                                                    TRAN3
          CALL  BGET                                                     TRAN3
      140 CONTINUE                                                       TRAN3
C     VII-B CHECK COMSTRINT VECTOR ELEMENTS                              SEP18
          IF(FTIME GT TST1(ISTN)+1.E-4) GO TO 170                        SEP18
          DO 143 I=1,NCNST                                               SEP18
      143 JJ(I)=1                                                        SEP18
          IF(INTB-1)170,150,160                                          SEP18
      150 IF(ISTN LT NSB) GO TO 170                                      SEP18
      151 LL=1                                                           SEP18
          MM=NICNB                                                       SEP18
          GO TO 165                                                      SEP18
      160 IF(ISTN LE.NSB) GO TO 170                                      SEP18
          IF(ISTN GT.NSB+NSAB) GO TO 151                                 SEP18
          LL=NICNB+1                                                     SEP18
          MM=NCNST                                                       SEP18
      165 DO 167 IJI=LL,MM                                               SEP18
      167 JJ(IJI)=0                                                      SEP18
      170 CONTINUE                                                       TRAN3
C          VIII-C          DO  TRANSFORMATION                            COMN
          DO 190 MA=1,NEQ                                                TRAN3
          DO 180 MP=1,NCNST                                              TRAN3
          IF(JJ(MP).EQ 0) GO TO 180                                      TRAN3
          XL(MA,NCN) = XL(MA,NCN)- TR(MP)*XL(MA,MP)                      TRAN3
      180 CONTINUE                                                       TRAN3
C       VIII-D  ACCOUNT FOR SIGN ON PAYOFF                               SEP18
          XL(MA,NCN)= XL(MA,NCN)*SIG                                     SEP18
      190 CONTINUE                                                       TRAN3
C          IX        TRANSFORM  IMPULSE  RESPONSE FUNCTIONS              COMN
          DO  200 MP=1,NCNST                                             TRAN3
          XLAMP(NCN)=XLAMP(NCN)-TR(MP)*XLAMP(MP)                         TRAN3
      200 XLAMA(NCN) =XLAMA(NCN) -TR(MP)*XLAMA(MP)                       TRAN3
C          X       WRITE   TRANS. DATA  ON  FILE                         COMN
          WRITE(10) (XL(J,NCN),J=1,NEQ),XLAMA(NCN) ,XLAMP(NCN)           TRAN3
C          XI        TEST  FOR  END  OF LAST  ARC   TO QUIT              COMN
          IF(ISTN.EQ NST AND ABS(FTIME-DIS1(NST)) LT 1 E-6) GO TO 220    TRAN3
      210 CONTINUE                                                       TRAN3
      220 RETURN                                                         TRAN3
C          XII     TRANSFORMATION  FOR  UNCONSTRAINED  PROPLEM           COMN
C                    IS IDENTITY MATRIX  TIME  SIGN  ON  PAYOFF          COMN
      230 DO 260 KJPOS=1,2000                                            TRAN3
          CALL AGETA                                                     TRAN3
          CALL  BGET                                                     TRAN3
      240 CONTINUE                                                       TRAN3
          DO 250 I=1,NEQ                                                 TRAN3
      250 XL(I,NCN) = XL(I,NCN)*SIG                                      TRAN3
          WRITE(10) (XL(J,NCN),J=1,NEQ),XLAMA(NCN) ,XLAMP(NCN)           TRAN3
          IF (ABS(FTIME-DIS1(NST)) LT.1 E-6) GO TO 220                   TRAN3
      260 CONTINUE                                                       TRAN3
      270 CALL STPIT(99)                                                 TRAN3
          END                                                            TRAN3
```

20 OCT 72 G.01-46

SUBRØUTINE
TRTØSZ

Subroutine TRTØSZ

Entry Points. INTER1, ARCEND, TRJEND

Purpose

Saves trajectory data required for sizing during solution trajectory integration. Entry INTER1 saves initial states. Entry ARCEND saves data at intermediate arc end points. Entry TRJEND saves final weights, impulsive velocity, and velocity losses.

Description

Entry INTER1 is called from FNTG at the initiation of the forward trajectory. Entry ARCEND is called by subroutine PRØPIN. Entry TRJEND is called by FNTG at the terminus of the solution trajectory.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DCON	$d\Psi_I$	I	Asked for correction in constraint misses and payoff vector	/GENF /(	289)	CON3	O	DCON
						MTX3A	I	DCON
						PAYO2	M	DCON
						TEST	M	DCON
						TOPM	I	DCON
						TRTOSZ	I	DCON
DPSQ	$(dP)^2$	I	Metric of control and parameter changes [sd]	/GENF /(	302)	PAYO2	M	DPSQ
						TEST	M	PSISQ
						TRTOSZ	I	DPSQ
DVO		O	Orbiter ideal velocity (fps)	/SIZING/(	307)	SIZOUT	I	DVO
						SIZ1	O	DVO
						SIZ2	O	DVO
						SIZ3	M	DVO
						SIZ4	O	DVO
						TRTOSZ	O	DVO
ER	E_R	I	Earth radius. (FT)	/GLOBAL/(	2)	COORDS	I	ER
						CRASH	I	REM
						EQUA3	I	ER
						GEINP	I	ER
						PADS1	I	ER
						PDBC	I	ER
						SOMG	I	ER
						TRTOSZ	I	ER
FTNM		I	Feet to naut mi conversion, 1 645791629x10^{-4}	/DATA /(	7)	OUT	I	FTNM
						PADS1	O	FTNM
						TRTOSZ	I	FTNM
IARC		I	Arc number	/XCODES/(	146)	ADICB3	I	IARC
						ADID3A	I	IARC
						ADJUST	I	IARC
						AST3	I	IARC
						BNTG	M	IARC
						ENVPRM	I	IARC
						FNTG	M	IARC
						GETIT	I	IARC
						MODELA	I	IARC
						PROPB	I	IARC
						PROPIN	I	IARC
						REU3	I	IARC
						SDINP	M	IARC
						STAU	I	IARC
						STP3	I	IARC
						TRTOSZ	I	IARC
IDVEL		O	Total ideal velocity required to orbit (fps)	/SIZING/(	297)	SIZ1	I	IDVEL
						SIZ2	I	IDVEL
						SIZ3	I	IDVEL
						SIZ4	I	IDVEL
						TRTOSZ	O	IDVEL
INTB		I	Branching and intermediate constraint flag	/XCODES/(	31)	ADIC3A	I	INTB
						BNTG	I	INTB
						ENVPRM	I	INTB
						FNTG	I	INTB
						SDINP	M	INTB
						TEST	I	INTB
						TRAN3	I	INTB
						TRTOSZ	I	INTB
IPFLG1		O	IPFLG1≠0 supresses print-out of velocity losses and inertial Euler angles	/GLOBAL/(	69)	FNTG	I	IPFLG1
						OUT	I	IPFLG1
						PDBC	I	IPFLG1
						PRINT	I	IPFLG1
						TRTOSZ	O	IPFLG1
IPFLG2		O	IPFLG2≠0 supresses print-out of orbital parameters	/GLOBAL/(	70)	PRINT	I	IPFLG2
						TRTOSZ	O	IPFLG2
IPFLG3		O	IPFLG3≠0 supresses print-out of impact data	/GLOBAL/(	71)	OUT	I	IPFLG3
						PRINT	I	IPFLG3
						TRTOSZ	O	IPFLG3

8 NOV 72 G 01-46

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
JTYP		I	Sizing. Flag	/SIZING/(	313)	FNTG	I	JTYP
						GEINP	O	JTYP
						MODELA	I	JTYP
						PADS1	I	JTYP
						PROPIN	I	JTYP
						SIZIN	I	JTYP
						TRTOSZ	I	JTYP
NCN		I	Number of elements in dY	/XCODES/(	160)	ADEQ3A	I	NCN
						ADICB3	I	NCN
						ADIC3A	I	NCN
						ADID3A	I	NCN
						ADJUST	I	NCN
						AST3	M	NCN
						BNTG	I	NCN
						BST03	I	NCN
						MTX3A	I	NCN
						OUT	I	NCN
						PAY02	M	NCN
						TEST	M	NCN
						TOPM	I	NCN
						TRAN3	I	NCN
						TRTOSZ	I	NCN
NSAB		I	Number of arcs on first branch	/XCODES/(	134)	ADICB3	I	NSAB
						BNTG	I	NSAB
						ENVPRM	I	NSAB
						FNTG	I	NSAB
						SDINP	M	NSAB
						TEST	I	NSAB
						TRAN3	I	NSAB
						TRTOSZ	I	NSAB
NSB		I	Number of arcs prior to branch point or intermediate constraint	/XCODES/(	133)	ADICB3	I	NSB
						BNTG	I	NSB
						ENVPRM	I	NSB
						FNTG	I	NSB
						REU3	I	NSB
						SDINP	M	NSB
						TEST	I	NSB
						TRAN3	I	NSB
						TRTOSZ	I	NSB
R	R	I	Radial distance from earth center to vehicle (FT)	/GENF—/(	305)	BL4	I	R
						BL7	I	R
						BL8	I	R
						DER3A	I	R
						EQUA3	M	R
						MODELA	I	R
						MODELB	I	R
						POBC	I	R
						POY3A	I	R
						TRTOSZ	I	R
RAD		I	Radian to angle conversion, 57 29577951	/DATA /(	2)	BEROCO	I	DEG
						BLGCON	I	RAD
						ENVPRM	I	RAD
						EQUA3	I	RAD
						FNTG	I	RAD
						GUI3A	I	RAD
						MODELA	I	RAD
						MTX3A	I	RAD
						OUT	I	RAD
						PADS1	D	RAD
						SDINP	I	RAD
						TRTOSZ	I	RAD

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SQ		M	A synthesis data array (37,5) that contains the flyback data and some injection quantities	/SIZING/(	74)	ENVPRM	M	SQ
						FLYBKP	M	SQ
						ISPRAT	I	SQ
						PDBC	I	SQ
						PRITVA	I	SQ
						RANGE	M	SQ
						REU3	O	SQ
						SIZE	O	SQ
						SIZEMR	M	SQ
						SIZIN	M	SQ
						STAU	I	SQ
						SUMOUT	M	SQ
						TAMPAR	O	SQ
						TAMPER	M	SQ
						THRUST	M	SQ
						TRTOSZ	M	SQ
						VEHDF	M	SQ
						WTVOL	M	SQ
STOT	S_T	I	Total range (FT)	/ORBIT /(	158)	OUT	I	STOT
						PDBC	O	STOT
						TRTOSZ	I	STOT
SV		M	A synthesis array (28) containing staging parameters and misc flags	/SIZING/(	46)	ENVPRM	M	SV
						FLYBKP	I	SV
						ITER8	I	SV
						RANGE	I	SV
						SIZEMR	M	SV
						SIZIN	I	SV
						SSSP	I	SV
						SUMOUT	I	SV
						TAMPAR	O	SV
						TAMPER	M	SV
						TRTOSZ	M	SV
						VEHDF	M	SV
						WTVOL	I	SV
SVAR	$y\|_{t=0}$	I	Array of state values at initial problem time [sd]	/GENF /(	79)	ADJUST	O	SVAR
						BNTG	I	SVAR
						FNTG	I	SVAR
						PRMSET	M	SVAR
						REU3	I	SVAR
						SDINP	M	SVAR
						TEST	I	SVAR
						TOPM	I	SVAR
						TRTOSZ	I	SVAR
SVDCON		O	Saved payoff improvement	/SIZING/(	320)	TRTOSZ	O	SVDCON
SVDPSQ		O	Saved control matric	/SIZING/(	319)	PAY02	I	SVDPSQ
						TRTOSZ	O	SVDPSQ
TST1		I	Arc end times for nominal trajectory	/GENF /(	433)	BNTG	I	TST1
						GETIT	I	TST1
						PROPIN	I	TST1
						SDINP	O	TST1
						TEST	O	TST1
						TOPM	I	TST1
						TRAN3	I	TST1
						TRTOSZ	I	TST1
VSTG		M	Booster staging velocity (fps)	/SIZING/(	311)	SIZE	I	VSTG
						TRTOSZ	M	VSTG
W	W	I	Weight (LBS)	/GENF /(	412)	BL5	I	W
						ENVPRM	I	W
						EQUA3	M	W
						FH3	I	W
						OUT	I	W
						PDBC	I	W
						REU3	I	W
						TRTOSZ	I	W

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
WFO		O	Orbiter burnout weight (lb)	/SIZING/(	296)	PAYLOD	M	WFO
						SIZOUT	I	WFO
						SIZ1	O	WFO
						SIZ2	O	WFO
						SIZ3	O	WFO
						SIZ4	M	WFO
						TAMPAR	I	WFO
						TRTOSZ	O	WFO

8 NOV 72 G 01-46

```
      SUBROUTINE  TRTOSZ                                                TRTOSZ
C          SAVES   TRAJECTORY-TO-SIZING DATA  DURING SOLUTION TRAJECTOR TRTOSZ
C          ENTRY  POINTS  INCLUDE                                       TRTOSZ
C          INTER1   SAVE  INIT. STATES                                  TRTOSZ
C          ARCEND   SAVES INTERMEDIATE  DATA                            TRTOSZ
C          TRJEND   SAVES  FINAL WTS. AND IMPULSIVE VELOCITY            TRTOSZ
C                                                                       TRTOSZ
      COMMON/GENF/                                                      GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)     ,SVAR(10)  ,WDC(20)  ,  GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9)  ,DCON(9)   ,DTP      ,  GENF
     *DTS        ,DT        ,G         ,DPSQ       ,Q         ,QS       ,  GENF
     *R          ,RE        ,MACH      ,PA         ,RO        ,CS       ,  GENF
     *VNU        ,PAR       ,ROR       ,CSR        ,VNR       ,SUMSQ    ,  GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP        ,TOS       ,TR(9)    ,  GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)    ,T         ,W        ,  GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)   ,TIME      ,DMP      ,  GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX        ,TBURN     ,TBU(20)  ,  GENF
     *AE         ,FP        ,FPOLD     ,FPD        ,MACHR     ,MACHV    ,  GENF
     *QR         ,QV        ,FVAC      ,LIFTV      ,                       GENF
     *LIFTR      ,LIFTA     ,           DRAGV      ,DRAGR     ,DRAGA    ,  GENF
     *            LIFTM     ,DBR       ,DB         ,ISP       ,ISPF     ,  GENF
     *                       ULFT      ,ULFTV      ,ULFTR     ,ULFTA    ,  GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA      ,XMCGM     ,CODAE    ,  GENF
     *CULFT      ,CT        ,CALPHA    ,CDE        ,DELTAE    ,SID      ,  GENF
     *COD        ,SIDAE     ,XCG       ,ZCG        ,XJ                     GENF
      COMMON / GENF  /                                                  GENF
     *XJV        ,XJR       ,GH        ,GAMMAD     ,XKG       ,XKP      ,  GENF
     *FRATED    ,IRATED    ,                                            GENF
     *P1         ,P2        ,P3       ,XK1         ,XK2       ,XK3      ,  GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D       ,XK2D      ,XK3D     ,  GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V       ,XK2V      ,XK3V     ,  GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P       ,XK2P      ,XK3P     ,  GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1D       ,XK2D      ,XK3D     ,  GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M       ,XK2M      ,XK3M     ,  GENF
     *PV         ,PG        ,PP        ,PR         ,PD        ,DPDY(3,8)   GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,                  MACH, MACHR,  GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                  FRAT
      DIMENSION TPH1(10),TST1(10)                                       GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                GENF
      REAL MUB, MUD, ISPB, ISPO, IDVEL,NNB,ND                           SIZING
      COMMON /SIZING/                                                   SIZING
C            PHASE II SIZING PARAMERERS                                 SIZING
     *TZ,        VV(3),      QP(14),    EROR,     PZ(5),      VQ,        SW(20), SIZING
     *SV(28),   SQ(37,5),  SE(11),    TLAT,      TLNG,                  SIZING
C           PHASE I SIZING PARAMERERS                                   SIZING
     *WBO,       WLOO,      DWEB,      DWEO,      TOLWT,    WPB,     TWRAT2, SIZING
     *BK1,       BK2,       BK3,       BK4,       ISIZE,    TRAFLG,  TWRATO, SIZING
     *OK1,       OK2,       OK3,       OK4,       PRFLG,    IPASS,   IPSMAX, SIZING
     *AEXIT,     TVACO,     ND,        WFO,       IDVEL,    ISPO,    ISPB,   SIZING
     *XPL,       TVACB,     NNB,       WEO,       WEB,      WO,      WLO,    SIZING
     *DVO,       DVB,       MUB,       MUD,       VSTG,     WPO              SIZING
     *,JTYP,     BECO ,     BSTG ,     ORBI,     ITNBW , ITNOW ,            SIZING
     * SVOPSQ  , SVOCON   ,IHUNT     ,IOPSTG   ,ISZD(19)                    UH
      COMMON /XCODES/                                                   XCODES
     *ITQ     (9),ICOR  (20),ITI       ,INTB       ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB        ,NSAB      ,NICNB    ,  XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR       ,IFB       ,IND      ,  XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST       ,IARC      ,ISTART   ,  XCODES
     *ITCT       ,ITER      ,IVAR      ,JK         ,JPS       ,JS       ,  XCODES
     *KOP        ,KPST      ,K         ,KST        ,NAD       ,NCASE    ,  XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP        ,NPH       ,N        ,  XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN       ,IPHN      ,ISTNB    ,  XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP      ,ISTPP     ,L        ,  XCODES
     *IFOB       ,NB        ,LB        ,MB         ,NPHP      ,NPHB     ,  XCODES
     *NCTIN      ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3          XCODES
      COMMON /PRINT/ AP(100)                                            TRTOSZ
      COMMON/ ORBIT/  VI,      GAMI,      PSII,      XMUI,        P,     ORBIT
     *      ECC,     AINCL,      ARGP,   ASCNOD,   SMIMAJ,   APOGEE,     ORBIT
     *  PERGEE,   ANOMLY,      CAPX,      CAPY,    ASYMP,   ENERGY,      ORBIT
     *   HMNTM,    DVIDV,    DVIDG,                                      ORBIT
     *  DVIDH ,    DVIDM,   DVIDPS,   DVIDRO,   DVIDMU,    DGIDV,        ORBIT
     *  DGIDG ,    DGIDH,   DGIDM ,   DGIDPS,   DGIDRO,    DGIDMU,       ORBIT
     *  DPIDV ,    DPIDG,   DPIDH ,   DPIDM ,   DPIDPS,    DPIDRO,       ORBIT
     *  DPIDMU,    DMIDV,   DMIDG ,   DMIDH ,   DMIDM ,    DMIDPS,       ORBIT
     *  DMIDRO,    DMIDMU,  DPDV  ,   DPDG  ,   DPDH  ,    DPDM  ,       ORBIT
```

```
      *  DPDPS ,     DPDRO ,  DPDMU ,    DECDV ,    DECDG ,     DECDH ,         ORBIT
      *  DECDM ,     DECDPS,  DECDRO,    DECDMU,    DIDV  ,     DIDG  ,         ORBIT
      *  DIDH  ,     DIDM  ,  DIDPS ,    DIDRO ,    DIDMU ,     DBEDV ,         ORBIT
      *  DBEDG ,     DBEDH ,  DBEDM ,    DBEDPS,    DBEDRO,     DBEDMU,         ORBIT
      *  DNODV ,     DNODG ,  DNODH ,    DNODM ,    DNODPS,     DNODRO,         ORBIT
      *  DNODMU,     DSMDV,     DSMDG,    DSMDH,     DSMDM,    DSMDPS           ORBIT
       COMMON/ORBIT/                                                           ORBIT
      *  DSMDRO,   DSMDMU,     DAPDV,     DAPDG,     DAPDH,    DAPDM ,         ORBIT
      *  DAPDPS,   DAPDRO,     DAPDMU,    DPEDV,     DPEDG,    DPEDH ,         ORBIT
      *DPEDM     ,DPEDPS     ,DPEDRO     ,DPEDMU     ,DANDV      ,DANDG      , ORBIT
      *  DANDH ,   DANDM ,     DANDPS,    DANDRO,    DANDMU,   DCXDV ,         ORBIT
      *  DCXDG ,   DCXDH ,     DCXDM ,    DCXDPS,    DCXDRO,   DCXDMU,         ORBIT
      *  DCYDV ,   DCYDG ,     DCYDH ,    DCYDM ,    DCYDPS,   DCYDRO,         ORBIT
      *  DCYDMU,   DASDV ,     DASDG ,    DASDH ,    DASDM ,   DASDPS,         ORBIT
      *  DASDRO,   DASDMU,     DENDV ,    DENDG ,    DENDH ,   DENDM ,         ORBIT
      *  DENDPS,   DENDRO,     DENDMU,    DMODV ,    DMODG ,   DMODH ,         ORBIT
      *  DMODM ,   DMODPS,     DMODRO,    DMODMU                               ORBIT
       DIMENSION ORBPRM(18),PPO(7,18)                                          ORBIT
       EQUIVALENCE (VI,ORBPRM),(DVIDV,PPO)                                     ORBIT
       COMMON/ORBIT/ YMXRF,SNXLMR,CSXLMR,SDOWN,SCROSS,TD,TC                    ORBIT
      *,SNPSR    ,CSPSR      ,SNGI       ,CSGI       ,SPSII      ,CPSII      , ORBIT
      * STOT     ,CSI        ,SNI        ,SNGNU      ,CSAND      ,COSDMU     , ORBIT
      * SINDMU   ,THT        ,WTFUEL                                           ORBIT
       COMMON/STATE3/                                                          STATE3D
      *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)       ,SVY(10)    , STATE3D
      *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM     ,SAVBP(15)  , STATE3D
      *SINPSI    ,COSPSI     ,SINRHO     ,COSRHO     ,OCORHO     ,OCORO2     , STATE3D
      *SVBV   (9),OMEGA      ,OMEGA2     ,                                     STATE3D
      *VDV       ,GDV        ,RDV        ,MDV        ,PDV        ,ODV        , STATE3D
      *UDV       ,VDG        ,GDG        ,RDG        ,PDG        ,ODG        , STATE3D
      *UDG       ,VDR        ,GDR        ,MDR        ,PDR        ,ODR        , STATE3D
      *UDR       ,VDM        ,GDM        ,MDM        ,PDM        ,VDP        , STATE3D
      *GDP       ,PDP        ,ODP        ,UDP        ,VDO        ,GDO        , STATE3D
      *PDO       ,UDO        ,HTDV       ,HTDR                                 STATE3D
       REAL MDM , MDV, MDR                                                     STATE3D
       COMMON/STATE3/                                                          STATE3D
      *SIN2RO    ,COS2RO     ,COS2GM                                           STATE3D
       EQUIVALENCE (VAR(1),V) ,(VAR(2),GAM) ,(VAR(3),ALT) ,(VAR(4),M) ,        EQUV3
      *(VAR(5),PSI),(VAR(6),RHO),(VAR(7),MU),(VAR(8),HT) , (VAR(9),SQ2),       EQUV3
      *(DVAR(1),VD),(DVAR(2),GD),(DVAR(3),HD),(DVAR(4),MD),(DVAR(5),PD),       EQUV3
      *(DVAR(6),OD),(DVAR(7),UD),(DVAR(8),HTD),(DVAR(9),SQ2D)                  EQUV3
       REAL M,MU,MD                                                            EQUV3
       COMMON/DATA/                                                            DATA
      *PI        ,RAD        ,RDI        ,SC         ,UMF        ,TMPF       , DATA
      *FTNM      ,CAR        ,JOP1       ,JOP2       ,JOP3       ,JOP4       , DATA
       COMMON/GLOBAL/                                                          GLOBAL
      *GR        ,ER         ,OMGZ       ,XLAMRF     ,YMURF      ,LUM          GLOBAL
      *,JJOP(10) ,IFATAL     ,NARC       ,NBRAN      ,NFARC      ,ID(4)        GLOBAL
      *,KTAB(20),ITAB(20)   ,SIG     ,MAXTAB                                   GLOBAL
      *,GM       ,PSIRF,IPFLG1,     IPFLG2,IPFLG3,IPFLG4,INEQFL(20)            GLOBAL
      *,ITPSO  ,KSOL       ,KGLOBL(8)                                          RETAP
C        I       INITIAL  STATE  SAVE                                          TRTOSZ
       ENTRY INTER1                                                            TRTOSZ
       SQ(17,1) =  SVAR(2)                                                     TRTOSZ
       IF(JTYP NE.2) GO TO 5                                                   UH        5
       IPFLG1=0                                                                UH
       IPFLG2=0                                                                UH
       IPFLG3=0                                                                UH
     5 CONTINUE                                                                UH
       SQ(17,2) =  SVAR(3)*RAD                                                 TRTOSZ
       SQ(17,3) =  SVAR(4)                                                     TRTOSZ
       SQ(17,4) =  SVAR(6)*RAD                                                 TRTOSZ
       SQ(17,5) =  SVAR(7)*RAD                                                 TRTOSZ
       SQ(18,1) =  SVAR(8)*RAD                                                 TRTOSZ
       SVDPSQ = DPSQ                                                           AAA
       SVDCON = DCON(NCN)                                                      AAA
       IF(INTB.NE 2) RETURN                                                    AAA
       SV(21)=1.                                                               AAA
       CALL ENVPRI                                                             AAA
       RETURN                                                                  TRTOSZ
C         II      INTERMEDIATE  ARC DATA                                       TRTOSZ
       ENTRY ARCEND                                                            TRTOSZ
       IM1 = IARC-1                                                            TRTOSZ
       IF(JTYP LE 0) GO TO 30                                                  AAA       30
C         II-A   PHASE I  SIZING  DATA                                         TRTOSZ
```

```
   10 CONTINUE                                                         TRTOSZ
C  II-B TEST  FOR  KEY  ARCS                                           TRTOSZ
C               BOOSTER  THRUST TERMINATION ARC                        TRTOSZ
      IF   (IFIX(SQ(1,1)).EQ.IM1) GO TO 100                            TRTOSZ
C               OPTIMAL  STAGE TIME                                    TRTOSZ
   20 IF( IFIX(SQ(1 ,2)) EQ.IM1)  GO TO 110                            TRTOSZ
C               OPTIMAL  PITCHOVER TIME                                TRTOSZ
   30 IF  (IFIX(SQ(13,3)) EQ IM1) GO TO  120                           TRTOSZ
C            TEST  FOR  BRANCHING  AND  INJECTION                      AAA
      IF(INTB EQ.2.AND.NSB+NSAB GE.IFIX(SQ(1,3)).AND IARC EQ NSB+NSAB) AAA
     * GO  TO 500                                                      AAA
C            TEST  FOR   BRANCHING  AND  ENTRY TERMINUS                AAA
      IF(INTB.EQ 2.AND.NSB+NSAB.LT.IFIX(SQ(1,3)).AND.IARC.EQ NSB+NSAB) AAA
     *GO  TO  600                                                      AAA
      RETURN                                                           TRTOSZ
C            BOOSTER   CHARACTERISTIC  VELOCITY                        TRTOSZ
  100 VSTG   =  AP(17)                                                 TRTOSZ
      IF(JTYP.EQ.1) GO TO 20                                           AAA
C      III-A  PHASE II  AT  STAGING  POINT                             AAA
      QP(9) =AP(9)                                                     AAA
      SW(10)=STOT/ER                                                   AAA
      SW(20)=AP(1)                                                     AAA
      SV(7)=W                                                          FIN1
      SV(8) =AP(3)                                                     AAA
      SV(9) =AP(2)                                                     AAA
      SV(10)=AP(4)                                                     AAA
      SV(12)=AP(32)                                                    AAA
      SV(17)=AP(5)                                                     AAA
      SV(18)=AP(7)                                                     AAA
      SV(19)=AP(6)                                                     AAA
      SV(21)= 0.                                                       AAA
      SV(22)= R                                                        AAA
      SV(23)= AP(30)                                                   AAA
      SV(24)= AP(29)                                                   AAA
      SV(25)= 0.                                                       AAA
      SV(26)= 90.- AP(92)                                              AAA
      SQ(12,1) = AP(77)/FTNM/ER                                        AAA
      SQ(36,1)= AP(55)                                                 AAA
      SQ(36,2)= AP(56)                                                 AAA
      SQ(37,1)= AP(1)                                                  AAA
      GO TO 20                                                         TRTOSZ
  110 SQ(18,2) = AP(8)                                                 TRTOSZ
      GO TO 150                                                        TRTOSZ
  120 SQ(18,3) = AP(8)                                                 TRTOSZ
  150 RETURN                                                           TRTOSZ
  300 CONTINUE                                                         TRTOSZ
      RETURN                                                           TRTOSZ
      ENTRY TRJEND                                                     TRTOSZ
      IF(JTYP-1)150, 310 400                                           TRTOSZ
  310 IDVEL= AP(17)                                                    TRTOSZ
      DVO  = AP(17) -VSTG                                              TRTOSZ
      WFO  = AP(9)                                                     TRTOSZ
      GO TO 150                                                        TRTOSZ
  400 CONTINUE                                                         AAA
      IF(INTB.NE.2) GO TO 500                                          AAA
      IF(IFIX(SQ(1,3))-NSB-NSAB)600,500,500                            AAA
C          PHASE II  INJECTION PT.  DATA                               AAA
  500 PZ(1) = AP(7)                                                    AAA
      PZ(2) = AP(6)                                                    AAA
      PZ(3) = AP(2)                                                    AAA
      PZ(4) = SVAR(6)*RAD -AP(5)                                       AAA
      PZ(5) = AP(20)                                                   AAA
      SV(7) = SQ(3,5)                                                  OS
      SV(3) = AP(17)                                                   AAA
      SV(4) = AP(9)                                                    AAA
      SV(5) = SV(7) -SV(4)                                             AAA
      SQ(3,1)=  AP(67)                                                 AAA
      SQ(3,2)=  AP(68)                                                 AAA
      SQ(3,3)=  AP(69)                                                 AAA
      SQ(14,4)= AP(10)                                                 AAA
      SQ(14,5)= AP(2)                                                  AAA
```

```
      SQ(15,3)= AP(11)                                                  AAA
      SQ(15,4)= AP(64)                                                  AAA
      SQ(36,3)= AP(55)                                                  AAA
      SQ(36,4)= AP(56)                                                  AAA
      SQ(36,5)= AP(57)                                                  AAA
C        ORBITER  BURN TIME                                             AAA
      IORBI = SQ(1,3)                                                   AAA
      SQ(37,2) = AP(1)-  TST1(IORBI)                                    AAA
      RETURN                                                            AAA
C        PHASE II  ENTRY  END POINT  (CRUISE RANGE)                     AAA
  600 SW(15) = AP(20)                                                   AAA
      RETURN                                                            AAA
      END                                                               TRTOSZ
```

25 OCT 72 8.01-46

SUBRØUTINE
VT

Subroutine VT

Entry. VTNOP

Purpose

Subroutine VT computes the control-dependent terms used in calculating the in-plane control vector. Entry VTNØP computes additional terms needed to calculate the partials of the governing equations with respect to the state.

Description

Subroutine VT is called by BLGCØN within its iterative scheme which solves for the in-plane control vector. This occurs on both forward trajectories and backward adjoint solutions. Entry VTNØP is called by BLGCØN only on backward adjoint solutions after the governing equations have been solved.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE		SUBROUTINE USAGE		
					BLOCK	LOC	SUBR	CODE	VAR
ALPHA	α	I	Angle of attack	(RAD)	/AECO3 /(	3)	BEROCO	I	ALPHA
							BLGCON	M	ALPHA
							BL2	I	ALPHA
							FNTG	O	ALPHA
							MAMECO	I	ALPHA
							MODELA	M	ALPHA
							MODELB	O	ALPHA
							REU3	O	ALPHA
							VT	I	ALPHA
CD	C_D	I	Drag coefficient		/AECO3 /(	52)	BEROCO	O	CD
							OUT	I	CD
							VT	I	CD
CDA	$\partial C_D/\partial\alpha$	I	See symbol		/AECO3 /(	53)	BEROCO	M	CDA
							VT	I	CDA
CDM	$\partial C_D/\partial M$	I	See symbol		/AECO3 /(	54)	BEROCO	O	CDM
							VT	I	CDM
CL	C_L	I	Lift coefficient		/AECO3 /(	49)	BEROCO	M	CL
							OUT	I	CL
							VT	I	CL
CLA	C_{L_α}	I	Lift coefficient slope		/AECO3 /(	50)	BEROCO	M	CLA
							EQUA3	M	CLA
							VT	I	CLA
CLM	$\partial C_L/\partial M$	I	See symbol		/AECO3 /(	51)	BEROCO	M	CLM
							VT	I	CLM
CM	C_M	I	Moment coefficient		/AECO3 /(	41)	MAMECO	O	CM
							VT	I	CM
CMA	C_{M_α}	I	Moment coefficient slope		/AECO3 /(	42)	EQUA3	M	CMA
							MAMECO	I	CMA
							VT	I	CMA
CMM	$\partial C_M/\partial M$	I	See symbol		/AECO3 /(	44)	MAMECO	O	CMM
							VT	I	CMM
COD	$\cos(\delta_E)$	M	See symbol		/GENF /(	556)	EL2	I	COD
							OUT	I	COD
							VT	M	COD
CODAE	$\cos(\alpha-\delta_E)$	O	See symbol		/GENF /(	549)	ACCEL	I	CODAE
							BL4	I	CODAE
							BL6	I	CODAE
							BL7	I	CODAE
							BL8	I	CODAE
							FH3	I	CODAE
							SDER3	I	CODAE
							VT	O	CODAE
COSA	$\cos\alpha$	M	See symbol		/AECO3 /(	8)	ACCEL	I	COSA
							BL4	I	COSA
							BL6	I	COSA
							BL7	I	COSA
							BL8	I	COSA
							FH3	I	COSA
							OUT	I	COSA
							VT	M	COSA
DB	D_b	I	Base drag	(LBS)	/GENF /(	537)	ACCEL	I	DB
							BL4	I	DB
							BL6	I	DB
							BL7	I	DB
							BL8	I	DB
							EQUA3	I	DB
							FH3	I	DB
							OUT	I	DB
							SDER3	I	DB
							VT	I	DB

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION		STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
DBR		I	Partial of base drag wrt altitude		/GENF	/(536)	ACCEL	I	DBR
							BL4	I	DBR
							BL6	I	DBR
							BL7	I	DBR
							BL8	I	DBR
							EQUA3	I	DBR
							FH3	I	DBR
							VT	I	DBR
DELTAE	δ_E	I	Engine gimbal deflection angle	(RAD)	/GENF	/(554)	BLGCON	M	DELTAE
							EL1	I	DELTAE
							OUT	I	DELTAE
							REU3	O	DELTAE
							VT	I	DELTAE
DRAG	D	M	Aerodynamic drag	(LBS)	/GENF	/(497)	ACCEL	I	DRAG
							BL5	I	DRAG
							BL7	I	DRAG
							BL8	I	DRAG
							ENVPRM	I	DRAG
							FH3	I	DRAG
							OUT	I	DRAG
							PROPB	O	DRAG
							PROPIN	O	DRAG
							SDER3	I	DRAG
							VT	M	DRAG
DRAGA		M	Partial of drag wrt angle of attack		/GENF	/(534)	ACCEL	I	DRAGA
							BL5	I	DRAGA
							BL7	I	DRAGA
							BL8	I	DRAGA
							FH3	I	DRAGA
							VT	M	DRAGA
DRAGR		M	Partial drag wrt altitude		/GENF	/(533)	ACCEL	I	DRAGR
							BL5	I	DRAGR
							BL7	I	DRAGR
							BL8	I	DRAGR
							FH3	I	DRAGR
							VT	M	DRAGR
DRAGV		M	Partial of drag wrt velocity		/GENF	/(532)	ACCEL	I	DRAGV
							BL5	I	DRAGV
							BL7	I	DRAGV
							BL8	I	DRAGV
							FH3	I	DRAGV
							VT	M	DRAGV
DREF	D_{ref}	I	Aerodynamic reference length		/ARCDAT/(	37)	VT	I	DREF
IATM		I	Atmosphere option flag		/ARCDAT/(	7)	EQUA3	I	IATM
							FXDAT	I	IATM
							OUT	I	IATM
							PDBC	I	IATM
							VT	I	IATM
JAER		I	Aerodynamic model option flag		/ARCDAT/(	9)	BEROCO	I	JAER
							EQUA3	I	JAER
							GEINP	I	JAER
							OUT	I	JAER
							PROPB	I	JAER
							PROPIN	I	JAER
							VT	I	JAER
LIFT	L	O	Aerodynamic lift	(LBS)	/GENF	/(496)	ACCEL	I	LIFT
							BL4	I	LIFT
							BL5	I	LIFT
							BL6	I	LIFT
							ENVPRM	I	LIFT
							FH3	I	LIFT
							OUT	I	LIFT
							PROPB	O	LIFT
							PROPIN	O	LIFT
							VT	O	LIFT

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
LIFTA		O	Partial of lift wrt angle-of attack	/GENF	/(531)	ACCEL	I	LIFTA
						BL4	I	LIFTA
						BL5	I	LIFTA
						BL6	I	LIFTA
						FH3	I	LIFTA
						VT	O	LIFTA
LIFTM		O	Partial of LIFT wrt mass	/GENF	/(535)	ACCEL	I	LIFTM
						BL4	I	LIFTM
						BL5	I	LIFTM
						BL6	I	LIFTM
						FH3	I	LIFTM
						VT	O	LIFTM
LIFTR		O	Partial of lift wrt altitude	/GENF	/(530)	ACCEL	I	LIFTR
						BL4	I	LIFTR
						BL5	I	LIFTR
						BL6	I	LIFTR
						FH3	I	LIFTR
						VT	O	LIFTR
LIFTV		O	Partial of lift wrt velocity	/GENF	/(529)	ACCEL	I	LIFTV
						BL4	I	LIFTV
						BL5	I	LIFTV
						BL6	I	LIFTV
						FH3	I	LIFTV
						VT	O	LIFTV
MACHR		I	Partial of mach number wrt altitude	/GENF	/(524)	EQUA3	O	MACHR
						PROPB	O	ZERO
						PROPIN	O	ZERO
						VT	I	MACHR
MACHV		I	Partial of mach number wrt velocity	/GENF	/(525)	EQUA3	O	MACHV
						VT	I	MACHV
Q		I	Dynamic pressure (PSF)	/GENF	/(303)	ENVPRM	I	Q
						EQUA3	M	Q
						OUT	I	Q
						PDBC	I	Q
						VT	I	Q
QR		I	Partial of dynamic pressure wrt altitude	/GENF	/(526)	EQUA3	M	QR
						VT	I	QR
QS		I	Product of dynamic pressure and aero Ref Area (LBS)	/GENF	/(304)	EQUA3	O	QS
						VT	I	QS
QV		I	Partial of dynamic pressure wrt velocity	/GENF	/(527)	EQUA3	M	QV
						TEST	M	QV
						VT	I	QV
SID	$\sin(\delta_E)$	M	See symbol	/GENF	/(555)	EL2	I	SID
						OUT	I	SID
						VT	M	SID
SIDAE	$\sin(\alpha-\delta_E)$	O	See symbol	/GENF	/(557)	ACCEL	I	SIDAE
						BL4	I	SIDAE
						BL6	I	SIDAE
						BL7	I	SIDAE
						BL8	I	SIDAE
						FH3	I	SIDAE
						VT	O	SIDAE
SINA	$\sin\alpha$	M	See symbol	/AECO3	/(7)	ACCEL	I	SINA
						BL4	I	SINA
						BL6	I	SINA
						BL7	I	SINA
						BL8	I	SINA
						FH3	I	SINA
						GUI3A	M	SINA
						OUT	I	SINA
						VT	M	SINA

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	LOC	SUBROUTINE USAGE SUBR	CODE	VAR
SREF	S_{ref}	I	Aerodynamic reference area (FT^2)	/ARCDAT/(	1)	BNTG EQUA3 FNTG FXDAT FXDAT GEINP SDINP SIZIN SIZIN THRUST VT	I I I I O M I I M I I	ARCDA SREF ARCDA ARCDA IARCDA ARCDA ARCDA ARCDA SREF SREF SREF
ULFT	L_U	M	Untrimmed aero.lift	/GENF /(	540)	BL3 MODELA VT	I I M	ULFT ULFT ULFT
ULFTA		M	Partial of ULFT wrt angle of attack	/GENF /(	543)	BL3 VT	I M	ULFTA ULFTA
ULFTR		M	Partial of ULFT wrt altitude	/GENF /(	542)	BL3 VT	I M	ULFTR ULFTR
ULFTV		M	Partial of ULFT wrt velocity	/GENF /(	541)	BL3 VT	I M	ULFTV ULFTV
XCG	X_{CG}	I	Center of gravity body x station (FT)	/GENF /(	558)	EL2 EQUA3 VT	I I I	XCG XCG XCG
XCGM	$\partial X_{CG}/\partial m$	I	See symbol	/AECO3 /(	38)	EL2 EQUA3 VT	I O I	XCGM XCGM XCGM
XCGR	X_{CGR}	I	Reference xcg location (FT)	/ARCDAT/(	32)	VT	I	XCGR
XJ	J	I	Control blend factor	/GENF /(	560)	EL2 EQUA3 OUT VT	I I I I	XJ XJ XJ XJ
XJR		I	Partial of blend factor wrt altitude	/GENF /(	562)	EL2 EQUA3 VT	I O I	XJR XJR XJR
XJV		I	Partial of blend factor wrt velocity	/GENF /(	561)	EL2 EQUA3 VT	I O I	XJV XJV XJV
XMCG	M_{CG}	M	Aerodynamic moment about center of gravity (FT-LBS)	/GENF /(	544)	EL2 OUT VT	I I M	XMCG XMCG XMCG
XMCGA		M	Partial of XMCG wrt angle of attack	/GENF /(	547)	EL2 VT	I M	XMCGA XMCGA
XMCGM		M	Partial of XMCG wrt mass	/GENF /(	548)	EL2 VT	I M	XMCGM XMCGM
XMCGR		M	Partial of XMCG wrt altitude	/GENF /(	546)	EL2 VT	I M	XMCGR XMCGR
XMCGV		M	Partial of XMCG wrt velocity	/GENF /(	545)	EL2 VT	I M	XMCGV XMCGV
XT	X_T	I	Aerodynamic trim surface body x station	/ARCDAT/(	36)	VT	I	XT
ZCG	Z_{CG}	I	Center of gravity body z station (FT)	/GENF /(	559)	EL2 EQUA3 VT	I I I	ZCG ZCG ZCG
ZCGM	$\partial Z_{CG}/\partial m$	I	See symbol	/AECO3 /(	39)	EL2 EQUA3 VT	I O I	ZCGM ZCGM ZCGM
ZCGR	Z_{CGR}	I	Reference zcg location (FT)	/ARCDAT/(	33)	VT	I	ZCGR
ZE	Z_E	I	Engine thrust centroid body z station	/ARCDAT/(	35)	EL2 VT	I I	ZE ZE

```
      SUBROUTINE VT                                                          VT
      REAL LCD, LSD, LCDA, LSDA                                              VT
      REAL LCDV, LCDR, LCDVV, LCDVR, LCDRR, LCDVA, LCDRA, LCDAA, LSDV,       VT
     *LSDR, LSDVV, LSDVR, LSDVA, LSDRR, LSDRA, LSDAA                         VT
      COMMON/ARCDAT/                                                         ARCDAT
     *SREF      ,EJ        ,XISP      ,TMULT     ,DTNC      ,DTPI     ,      ARCDAT
     *IATM      ,IMODE     ,JAER      ,JPRO      ,QMAX      ,GMAX     ,      ARCDAT
     *XLMAX     ,HDMAX     ,GMDOT     ,ALFMAX    ,PHMAX     ,MAEA     ,      ARCDAT
     *MAEB      ,MAEC      ,MAED      ,MAEE      ,MAEF      ,MAEG     ,      ARCDAT
     *MT        ,MISP      ,MXCG      ,MZCG      ,MWDA      ,MWDB     ,      ARCDAT
     *MDB       ,XCGR      ,ZCGR      ,XE        ,ZE        ,XT       ,      ARCDAT
     *DREF      ,MCND      ,RHOB      ,QMULT     ,REMAX                      ARCDAT
     * ,FRATE  , ARCD(9)                                                     RETAP
      DIMENSION  ARCDA(40)                                                   ARCDAT
      EQUIVALENCE(SREF,ARCDA)                                                ARCDAT
      COMMON/GENF/                                                           GENF
     *OMG(20)   ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  ,      GENF
     *A(9,9)    ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      ,      GENF
     *DTS       ,DT        ,G         ,DPSQ      ,Q         ,QS       ,      GENF
     *R         ,RE        ,MACH      ,PA        ,RO        ,CS       ,      GENF
     *VNU       ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    ,      GENF
     *SVSQ      ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    ,      GENF
     *TST(20)   ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        ,      GENF
     *TLP1(20)  ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      ,      GENF
     *TIMPR     ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  ,      GENF
     *AE        ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    ,      GENF
     *QR        ,QV        ,FVAC      ,LIFTV     ,                           GENF
     *LIFTR     ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    ,      GENF
     *           LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     ,      GENF
     *                      ULFT      ,ULFTV     ,ULFTR     ,ULFTA    ,      GENF
     *XMCG      ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    ,      GENF
     *CULFT     ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      ,      GENF
     *COD       ,SIDAE     ,XCG       ,ZCG       ,XJ                         GENF
      COMMON / GENF  /                                                       GENF
     *XJV       ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      ,      GENF
     *FRATED    ,IRATED    ,                                                 GENF
     *P1        ,P2        ,P3        ,XK1       ,XK2       ,XK3      ,      GENF
     *XK1T      ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     ,      GENF
     *XK1A      ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     ,      GENF
     *XK1G      ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     ,      GENF
     *XK1R      ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     ,      GENF
     *XK1U      ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     ,      GENF
     *PV        ,PG        ,PP        ,PR        ,PO        ,DPDY(3,6)        GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,              MACH, MACHR,           GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                       FRAT
      DIMENSION TPH1(10),TST1(10)                                            GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                                     GENF
      COMMON/STATE3/                                                         STATE3D
     *VAR(14)   ,DVAR  (14),VARL (99) ,DVARL(99) ,YD(9)     ,SVY(10)  ,      STATE3D
     *XL(9,9)   ,YDP(20,9),YDS (20,9),COSGAM     ,SINGAM    ,SAVBP(15) ,     STATE3D
     *SINPSI    ,COSPSI    ,SINRHO    ,COSRHO    ,OCORHO    ,OCORO2   ,      STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                      STATE3D
     *VDV       ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      ,      STATE3D
     *UDV       ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      ,      STATE3D
     *UDG       ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      ,      STATE3D
     *UDR       ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      ,      STATE3D
     *GDP       ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      ,      STATE3D
     *PDO       ,UDO       ,HTDV      ,HTDR                                  STATE3D
      REAL MDM , MDV, MDR                                                    STATE3D
      COMMON/STATE3/                                                         STATE3D
     *SIN2RO    ,COS2RO    ,COS2GM                                           STATE3D
      COMMON/AEC03/                                                          AEC03
     *APHD      ,APHR      ,ALPHA     ,VDA       ,GDA       ,PDA      ,      AEC03
     *SINA      ,COSA      ,PHIO      ,PHID      ,PHI       ,SINPHI   ,      AEC03
     *COSPHI    ,GDPH      ,PDPH      ,XLAMA(9)  ,XLAMP(9)  ,CDO      ,      AEC03
     *CDOM      ,CLO       ,FK        ,XCGM      ,ZCGM      ,CLOM     ,      AEC03
     *CM        ,CMA       ,CMAM ,CMM ,CMO       ,CMOM      ,FKM      ,      AEC03
     *CLAM      ,CL        ,CLA       ,CLM       ,                           AEC03
     *CD        ,CDA       ,CDM                                              AEC03
      ASSIGN 1041 TO LABL3                                                   VT
      ASSIGN 1071 TO LABL4                                                   VT
      ASSIGN 1061 TO LABL5                                                   VT
      GO TO 100                                                              VT   100
      ENTRY VTNOP                                                            VT
      ASSIGN 106 TO LABL3                                                    VT
```

20 OCT 72 G 01-46

```
      ASSIGN 109 TO LABL4                                        VT
      ASSIGN 1063 TO LABL5                                       VT
  100 SINA = SIN(ALPHA)                                          VT
      COSA = COS(ALPHA)                                          VT
      SID = SIN(DELTAE)                                          VT
      COD = COS(DELTAE)                                          VT
      SIDAE = SINA*COD - COSA*SID                                VT
      CODAE = COSA*COD + SINA*SID                                VT
  104 IF(SREF  LE  0. .OR  IATM  EQ. 2) RETURN                   VT
      CALL BEROCO                                                IO
      GO TO LABL3                                                VT
 1041 QMV = Q*MACHV                                              VT
      QMR = Q*MACHR                                              VT
      ULFTV  = SREF*(QV*CL + QMV*CLM)                            VT
      DRAGV  = SREF*(QV*CD + QMV*CDM)                            VT
      ULFTR  = SREF*(QR*CL + QMR*CLM)                            VT
      DRAGR  = SREF*(QR*CD + QMR*CDM)                            VT
  106 ULFTA = QS*CLA                                             VT
      DRAGA = QS*CDA                                             VT
      ULFT = QS*CL                                               VT
      DRAG = QS*CD                                               VT
      IF(JAER .EQ. 3) GO TO 107                                  VT    107
      GO TO LABL5                                                VT
 1061 CONTINUE                                                   VT
      LIFTV  = ULFTV                                             VT
      LIFTR  = ULFTR                                             VT
 1063 LIFTA  = ULFTA                                             VT
      LIFT = ULFT                                                VT
      RETURN                                                     VT
  107 CALL MAMECO                                                MAR14
      QSD = QS*DREF                                              VT
      LCD = ULFT*COSA + DRAG*SINA                                VT
      LSD = ULFT*SINA - DRAG*COSA                                VT
      LCDA = ULFTA*COSA + DRAGA*SINA                             VT
      LSDA = ULFTA*SINA - DRAGA*COSA                             VT
      XF = XCG - XCGR                                            VT
      ZF = ZCG - ZCGR                                            VT
      YF = ZE - ZCG                                              VT
      WF = XT - XCG                                              VT
      FACTOR = XJ/WF                                             VT
      LCDAA = LCDA - LSD                                         SEP18
      LSDAA = LSDA + LCD                                         SEP18
      XMCGA = LCDAA*XF + LSDAA*ZF + QSD*CMA                      VT
      XMCG = LCD*XF + LSD*ZF + DB*YF + QSD*CM                    VT
      GO TO LABL4                                                VT
 1071 SD = SREF*DREF                                             VT
      LCDV = COSA*ULFTV + SINA*DRAGV                             VT
      LCDR = COSA*ULFTR + SINA*DRAGR                             VT
      LSDV = SINA*ULFTV - COSA*DRAGV                             VT
      LSDR = SINA*ULFTR - COSA*DRAGR                             VT
      XMCGV = SD*(QV*CM + QMV*CMM) + LCDV*XF + LSDV*ZF           VT
      XMCGR = SD*(QR*CM + QMR*CMM) + LCDR*XF + LSDR*ZF + DBR*YF  VT
      XMCGM = LCD*XCGM + (LSD - DB)*ZCGM                         VT
      XCGMWF = XCGM/WF                                           VT
      FACTV = XJV/WF                                             VT
      FACTR = XJR/WF                                             VT
      FACTM = FACTOR*XCGMWF                                      VT
      LIFTV = FACTOR*XMCGV + FACTV*XMCG + ULFTV                  VT
      LIFTR = FACTOR*XMCGR + FACTR*XMCG + ULFTR                  VT
      LIFTM = FACTOR*XMCGM + FACTM*XMCG                          VT
  109 LIFTA = ULFTA + FACTOR*XMCGA                               VT
      LIFT = ULFT + FACTOR*XMCG                                  VT
      RETURN                                                     VT
      END                                                        VT
```

SUBRØUTINE
YREF3

Subroutine YREF 3

Entry Points. YREF, PSD

Purpose

Subroutine YREF3 (ENTRY YREF) refines cut-off time and state using third-order interpolation.

Entry PSD saves derivative at estimated cut-off point for use in entry YREF.

Description

The equations and logic for the subroutine are described in Section 15.1 of Volume I. This routine is called for non-time arc cut-offs from FNTG.

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	SUBROUTINE USAGE CODE	SUBROUTINE USAGE VAR
DT		O	Integration interval (SEC)	/GENF /(	300)	BNTG	M	DT
						FNTG	M	DT
						REU3	I	DT
						RKTA3A	I	P
						RKTB3A	I	P
						STP3	I	DT
						YREF3	O	DT
DVAR	$\dot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT
FP		I	Current value of cut-off function - non-linear only	/GENF /(	521)	DTF3	I	FP
						STP3	I	FP
						TOL3	I	FP
						YREF3	I	FP
FPD		I	Rate of change of non-linear cut-off function	/GENF /(	523)	CON3	I	FPD
						DTF3	I	FPD
						STP3	I	FPD
						YREF3	I	FPD
FPOLD		I	Value of non-linear cut-off function at prior compute interval	/GENF /(	522)	DTF3	I	FPOLD
						STP3	O	FPOLD
						TOL3	O	FPOLD
						YREF3	I	FPOLD
NEQ		I	Number of integrated states	/XCODES/(	162)	ADICB3	I	NEQ
						ADIC3A	I	NEQ
						ADID3A	I	NEQ
						AGETB3	I	NEQ
						AST3	I	NEQ
						BGET3	I	NEQ
						BSTO3	I	NEQ
						MTX3A	I	NEQ
						OUT	I	NEQ
						REU3	I	NEQ
						SDER3	I	NEQ
						SDINP	M	NEQ
						TOPM	I	NEQ
						TRAN3	I	NEQ
						YREF3	I	NEQ
SVY	y_{-1}	I	State and time array at previous compute interval	/STATE3/(	236)	DTF3	I	SVY
						REU3	M	SVY
						YREF3	I	SVY
TIME	t	M	Time (elapsed)	/GENF /(	493)	ADICB3	O	TIME
						AST3	I	TIME
						BNTG	M	TIME
						CON3	I	TIME
						DTF3	I	TIME
						ENVPRM	I	TIME
						EQUA3	I	TIME
						FNTG	M	TIME
						MODELA	I	TIME
						OUT	I	TIME
						PDBC	I	TIME
						PROPIN	I	TIME
						REU3	M	TIME
						RKTA3A	M	TT
						RKTB3A	M	TT
						YREF3	M	TIME

FORTRAN SYMBOL	MATH SYMBOL	CODE	DESCRIPTION	STORAGE BLOCK	STORAGE LOC	SUBROUTINE USAGE SUBR	CODE	VAR
V	v	M	Relative velocity (FT/SEC)	/STATE3/(	1)	ACCEL	I	V
						ADICB3	O	VAR
						ADJUST	M	VAR
						AGETB3	O	VAR
						AST3	I	VAR
						BL4	I	V
						BL7	I	V
						BL8	I	V
						CON3	I	VAR
						DER3A	I	V
						DTF3	I	V
						ENVPRM	I	VAR
						EQUA3	I	V
						MODELA	I	V
						MODELA	I	VAR
						MODELB	I	V
						MTX3A	I	VAR
						OUT	I	V
						OUT	I	VAR
						PDBC	I	V
						PDY3A	I	V
						REU3	M	VAR
						RKTA3A	M	Y
						STP3	I	VAR
						TOPM	O	KWOW
						YREF3	M	V
VT	$\dot{y}$	I	State vector derivatives in steepest descent module	/STATE3/(	15)	ADICB3	M	DVAR
						ADIC3A	I	DVAR
						ADID3A	M	DVAR
						DER3A	O	VD
						DTF3	I	VT
						ENVPRM	I	DVAR
						PDBC	I	VD
						PROPIN	O	DVAR
						REU3	I	DVAR
						RKTA3A	I	DY
						SDER3	O	DVAR
						STP3	I	DVAR
						YREF3	I	DVAR
						YREF3	I	VT

YREF3

```
      SUBROUTINE YREF3(II,OM)                                          YREF3
C                                                                      COMN
C        REFINES  CUT-OFF  TIME AND  STATE                             COMN
C              USING   THIRD ORDER  INTERPOLATION                      COMN
C                                                                      COMN
      COMMON/STATE3/                                                   STATE3D
     *VAR(14)    ,DVAR  (14),VARL (99) ,DVARL(99) ,YO(9)     ,SVY(10)  , STATE3D
     *XL(9,9)    ,YDP(20,9),YDS (20,9),COSGAM    ,SINGAM    ,SAVBP(15) , STATE3D
     *SINPSI     ,COSPSI    ,SINRHO    ,COSRHO    ,DCORHO    ,DCORD2   , STATE3D
     *SVBV   (9),OMEGA     ,OMEGA2    ,                                STATE3D
     *VDV        ,GDV       ,RDV       ,MDV       ,PDV       ,ODV      , STATE3D
     *UDV        ,VDG       ,GDG       ,RDG       ,PDG       ,ODG      , STATE3D
     *UDG        ,VDR       ,GDR       ,MDR       ,PDR       ,ODR      , STATE3D
     *UDR        ,VDM       ,GDM       ,MDM       ,PDM       ,VDP      , STATE3D
     *GDP        ,PDP       ,ODP       ,UDP       ,VDO       ,GDO      , STATE3D
     *PDO        ,UDO       ,HTDV      ,HTDR                             STATE3D
      REAL MDM , MDV, MDR                                              STATE3D
      COMMON/STATE3/                                                   STATE3D
     *SIN2RO     ,COS2RO    ,COS2GM                                    STATE3D
      COMMON/GENF/                                                     GENF
     *OMG(20)    ,OMGP(20,2),VARQ(9)   ,TOL(9)    ,SVAR(10)  ,WDC(20)  , GENF
     *A(9,9)     ,ACON(9)   ,BCON(9)   ,COTI(9,9) ,DCON(9)   ,DTP      , GENF
     *DTS        ,DT        ,G         ,DPSQ      ,Q         ,QS       , GENF
     *R          ,RE        ,MACH      ,PA        ,RO        ,CS       , GENF
     *VNU        ,PAR       ,ROR       ,CSR       ,VNR       ,SUMSQ    , GENF
     *SVSQ       ,TIMEPH    ,TIMES     ,TOP       ,TOS       ,TR(9)    , GENF
     *TST(20)    ,TPH   (20),DIS(20)   ,DIP(20)   ,T         ,W        , GENF
     *TLP1(20)   ,TLS1  (20),DIP1(20)  ,DIS1(20)  ,TIME      ,OMP      , GENF
     *TIMPR      ,LIFT      ,DRAG      ,TAX       ,TBURN     ,TBU(20)  , GENF
     *AE         ,FP        ,FPOLD     ,FPD       ,MACHR     ,MACHV    , GENF
     *QR         ,QV        ,FVAC      ,LIFTV     ,                      GENF
     *LIFTR      ,LIFTA     ,           DRAGV     ,DRAGR     ,DRAGA    , GENF
     *            LIFTM     ,DBR       ,DB        ,ISP       ,ISPF     , GENF
     *                       ULFT      ,ULFTV     ,ULFTR     ,ULFTA    , GENF
     *XMCG       ,XMCGV     ,XMCGR     ,XMCGA     ,XMCGM     ,CODAE    , GENF
     *CULFT      ,CT        ,CALPHA    ,CDE       ,DELTAE    ,SID      , GENF
     *COD        ,SIDAE     ,XCG       ,ZCG       ,XJ                    GENF
      COMMON / GENF  /                                                 GENF
     *XJV        ,XJR       ,GH        ,GAMMAD    ,XKG       ,XKP      , GENF
     *FRATED     ,IRATED    ,                                          GENF
     *P1         ,P2        ,P3       ,XK1        ,XK2       ,XK3      , GENF
     *XK1T       ,XK2T      ,XK3T      ,XK1D      ,XK2D      ,XK3D     , GENF
     *XK1A       ,XK2A      ,XK3A      ,XK1V      ,XK2V      ,XK3V     , GENF
     *XK1G       ,XK2G      ,XK3G      ,XK1P      ,XK2P      ,XK3P     , GENF
     *XK1R       ,XK2R      ,XK3R      ,XK1O      ,XK2O      ,XK3O     , GENF
     *XK1U       ,XK2U      ,XK3U      ,XK1M      ,XK2M      ,XK3M     , GENF
     *PV         ,PG        ,PP        ,PR        ,PO        ,DPDY(3,8)  GENF
      REAL  LIFTR , LIFT , LIFTA,LIFTM ,             MACH, MACHR,      GENF
     * ISP, ISPF, MACHV,LIFTV , IRATED                                 FRAT
      DIMENSION TPH1(10),TST1(10)                                      GENF
      EQUIVALENCE(TLP1,TPH1),(TLS1,TST1)                               GENF
      COMMON /XCODES/                                                  XCODES
     *ITQ    (9),ICOR  (20),ITI        ,INTB      ,JGID(20,2),JPH (20,2), XCODES
     *JST   (20)            ,NCNST     ,NSB       ,NSAB      ,NICNB    , XCODES
     *I2OP       ,ICOP      ,IFAW      ,IFAR      ,IFB       ,IND      , XCODES
     *IOPEN      ,IPH       ,ISPH      ,ISST      ,IARC      ,ISTART   , XCODES
     *ITCT       ,ITER      ,IVAR      ,JK        ,JPS       ,JS       , XCODES
     *KOP        ,KPST      ,K         ,KST       ,NAD       ,NCASE    , XCODES
     *NCN        ,NEQB      ,NEQ       ,NOP       ,NPH       ,N        , XCODES
     *NST        ,IPST      ,IPRINT    ,ISTN      ,IPHN      ,ISTNB    , XCODES
     *IPHNB      ,IBLK1     ,IBLK2     ,ISTOP     ,ISTPP     ,L        , XCODES
     *IFOB       ,NB        ,LB        ,MB        ,NPHP      ,NPHB     , XCODES
     *NCTIN      ,NEOF ,ILAB(8),JPRP,JGII,MTT,MPIN(20),JP1,JP2,JP3         XCODES
      COMMON/DATA/                                                     DATA
     *PI         ,RAD       ,RDI       ,SC        ,UMF       ,TMPF     , DATA
     *FTNM       ,CAR       ,JOP1      ,JOP2      ,JOP3      ,JOP4       DATA
      DIMENSION V(1),VT(4),SVDY(9)                                     YREF3
      DIMENSION SX(8)                                                  YREF3
      EQUIVALENCE(VAR,V),(DVAR,VT)                                     YREF3
      ENTRY YREF                                                       YREF3
      I= II                                                            YREF3
C                                                                      COMN
C      I   IF  STATE  CUT OFF   COMPUTE  SPACING AND  SET  DERIVATIVES COMN
C              ELSE  GO  TO  III                                       COMN
      IF(I.GT.7) GO TO 30                                              YREF3    30
```

20 OCT 72 G 01-46

```
      H= V(I)- SVY(I+1)                                         YREF3
      HP=  DM  - SVY(I+1)                                       YREF3
      ZDN = SVDY(I)                                             YREF3
      ZDN1 = VT (I)                                             YREF3
   10 REZ=HP/H                                                  YREF3
      R2= REZ*REZ                                               YREF3
      B=  R2*(3.-2.*REZ)                                        YREF3
      AA=1.-B                                                   YREF3
      C=  REZ*(REZ-1.)**2                                       YREF3
      D= R2 *(REZ-1.)                                           YREF3
      TIME = AA*SVY(1) +B*TIME +H*(C/ZDN + D/ZDN1)              YREF3
      DT =     TIME - SVY(1)                                    YREF3
      DO 20 IK= 1,NEQ                                           YREF3
      YDN= SVDY(IK)/ZDN                                         YREF3
      YDN1=  VT(IK)/ZDN1                                        YREF3
      V(IK)=AA*SVY(IK+1) + B*V(IK) + H*(C*YDN +D*YDN1)          YREF3
   20 CONTINUE                                                  YREF3
      RETURN                                                    YREF3
C                                                               COMN
C     III  FUNCTION  CUT-OFF   COMPUTE   SPACING  AND  DERIVATIVES COMN
   30 H= FP-FPOLD                                               YREF3
      HP=DM -FPOLD                                              YREF3
      ZDN= FPD                                                  YREF3
      MD=0                                                      YREF3
      I=II+2                                                    YREF3
      CALL PDBC (I,DVAR,SX,FPD,3,MD)                            YREF3
      ZDN1= FPD                                                 YREF3
      GO TO 10                                                  YREF3
C                                                               COMN
C     IV    ENTRY  USED   TO   STORE  DERIVATIVE  AT  ESTIMATED COMN
C             CUT-OFF  POINT  FOR  USE  IN  YREF   ABOVE        COMN
      ENTRY  PSD                                                YREF3
      DO 40 KI=1,NEQ                                            YREF3
   40 SVDY(KI)= VT(KI)                                          YREF3
      IF(II.LT.7) GO TO 50                                      YREF3
      MD=0                                                      YREF3
      I=II+2                                                    YREF3
      CALL PDBC (I,DVAR,SX,FPD,3,MD)                            YREF3
   50 CONTINUE                                                  YREF3
      RETURN                                                    YREF3
      END                                                       YREF3
```

10

50

20 OCT 72 6.01-46

4

PADS Main Program 2

103

Input Editor (INEDIT) **3**

167

156

161

124

Input Scan (GEINP) **4**